

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ROZPOZNÁNÍ POČASÍ Z KAMEROVÝCH SNÍMKŮ/SEKVENCÍ

WEATHER RECOGNITION FROM CAMERA IMAGES/SEQUENCES

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Juraj Sadel

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Ilona Janáková, Ph.D.

BRNO 2021

Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Juraj Sadel

ID: 197640

Ročník: 2

Akademický rok: 2020/21

NÁZEV TÉMATU:

Rozpoznání počasí z kamerových snímků/sekvencí

POKYNY PRO VYPRACOVÁNÍ:

Úkolem studenta je navrhnout algoritmy pro monitorování počasí z kamerových snímků, respektive sekvencí snímků. Předpokládá se detekce srážek (déšť, sněžení, krupobití) a snížené viditelnosti (mlhy) včetně parametrizace (např. odhad intenzity srážek ve scéně, odhad míry viditelnosti). Práce by se také měla zaměřit na detekci sněhu či vody na vozovce. Předpokládá se vymezení práce na dopravní scéně.

1. Seznamte se s danou problematikou. Provedte rešerši existujících přístupů.
2. Pořiďte dostatečně rozsáhlou a pestrou databázi reálných snímků/sekvencí.
3. Navrhněte možné přístupy detekce srážek a mlhy ve snímcích. V úvahu berte i výpočetní a časovou náročnost jednotlivých přístupů.
4. Zaměřte se na detekci srážek (sníh, voda) na vozovce.
5. Zvolené postupy implementujte a otestujte.
6. Definujte omezující podmínky. Zhodnoťte.

DOPORUČENÁ LITERATURA:

ŠONKA, Milan, Václav HLAVÁČ a Roger BOYLE. Image processing, analysis, and machine vision. Fourth edition. Stamford, CT: Cengage Learning, [2015]. ISBN 11-335-9360-7.

Termín zadání: 8.2.2021

Termín odevzdání: 17.5.2021

Vedoucí práce: Ing. Ilona Janáková, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

ABSTRAKT

Cieľom mojej diplomovej práce je navrhnúť a následne implementovať vhodné algoritmy klasifikácie počasia so zameraním sa hlavne na detekciu zrážok a hmly, vrátane parametrizácie. Prvá polovica práce je venovaná teoretickému popisu počasia a jeho vplyvom na dopravu. Ďalej je priblížená teória spracovania obrazu a neurónových sietí a rešerš existujúcich riešení. Následne sú popísané použité datasety. Praktická časť práce sa venuje návrhu možných algoritmov na základe teoretickej časti práce. Po návrhu sú jednotlivé algoritmy implementované, otestované a vyhodnotené. Na záver je popísané porovnanie klasických metód a neurónových sietí.

KĽÚČOVÉ SLOVÁ

počasie, doprava, spracovanie obrazu, počítačové videnie, neurónové siete

ABSTRACT

The aim of my master's thesis is to design and then implement suitable weather classification algorithms mainly focused on the detection of precipitation and fog, including parameterization. The first half of my thesis is devoted to a theoretical description of weather and its impact on transport. Furthermore, the theory of image processing and neural networks and already existing solutions is approached. Subsequently, the used datasets are described. The practical part of thesis is devoted to the design of possible algorithms based on the theoretical part of the thesis. After the design, the individual algorithms are implemented, tested and evaluated. Finally, a comparison of classical methods and neural networks is described.

KEYWORDS

weather, transport, image processing, computer vision, neural networks

SADEL, Juraj. *Rozpoznání počasí z kamerových snímků/sekvencí*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2021, 112 s. Diplomová práce. Vedúci práce: Ing. Ilona Janáková, Ph.D.

Vyhlásenie autora o pôvodnosti diela

Meno a priezvisko autora: Bc. Juraj Sadel
VUT ID autora: 197640
Typ práce: Diplomová práce
Akademický rok: 2020/21
Téma závěrečné práce: Rozpoznání počasí z kamerových snímků/sekvencí

Vyhlasujem, že svoju záverečnú prácu som vypracoval samostatne pod vedením vedúcej/cého záverečnej práce, s využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej záverečnej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto záverečnej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákonníka Českej republiky č. 40/2009 Sb.

Brno
.....
podpis autora*

*Autor podpisuje iba v tlačenej verzii.

POĎAKOVANIE

Rád by som poďakoval vedúcej diplomovej práce pani Ing. Ilone Janákovej, Ph.D. za odborné vedenie, konzultácie, ochotu, trpezlivosť a podnetné návrhy pri spracovávaní mojej diplomovej práce.

Obsah

Úvod	14
1 Úvod do problematiky	16
1.1 Počasie	16
1.1.1 Teplota a jej meranie	16
1.1.2 Atmosférický tlak a jeho meranie	17
1.1.3 Vietor a jeho meranie	18
1.1.4 Vlhkosť a jej meranie	19
1.1.5 Zrážky a ich meranie	20
1.1.6 Viditeľnosť a jej meranie	22
1.1.7 Oblačnosť a jej meranie	23
1.2 Počasie a doprava	24
2 Spracovanie obrazu	28
2.1 Digitalizácia	29
2.2 Predspracovanie	29
2.3 Segmentácia	29
2.3.1 Prahovanie	30
2.3.2 Segmentácia založená na detekcii hrán	30
2.3.3 Houghova transformácia	33
2.4 Popis a klasifikácia	34
2.5 Morfologické operácie	34
2.5.1 Erózia	35
2.5.2 Dilatácia	35
2.6 Analýza obrazu vo frekvenčnej oblasti	36
3 Neurónová sieť	38
3.1 Vrstvy neurónovej siete	38
3.2 Aktivačné funkcie	39
3.3 Stratová funkcia (Loss function)	41
3.4 Proces trénovania NN	42
3.5 Konvolúčna neurónová sieť	42
3.5.1 Konvolúčna vrstva	43
3.5.2 Združovacia (pooling) vrstva	44
3.5.3 Plne-prepojená vrstva (Fully-Connected)	44
3.5.4 VGGNet	45
3.6 Rekurentné neurónové siete RNN	45

3.7	Reziduálne neurónové siete ResNet	46
4	Existujúce riešenia	48
4.1	Detekcia zrážok vo vzduchu	49
4.1.1	Prvá metóda detekcie zrážok	49
4.1.2	Druhá metóda detekcie zrážok	50
4.2	Detekcia hmly	51
4.2.1	Prvá metóda detekcie hmly	51
4.2.2	Druhá metóda detekcie hmly	52
4.2.3	Tretia metóda detekcie hmly	52
4.3	Viac-triedové rozpoznávanie počasia	53
5	Vstupné dáta	55
5.1	Dataset stav vozovky	55
5.2	Dataset počasia s obecnou scénou	55
5.3	Dataset počasia s dopravnou scénou	57
6	Návrh a realizácia použitých algoritmov	58
6.1	Rozbor možných prístupov	58
6.1.1	Segmentácia vozovky a určenie stavu vozovky	58
6.1.2	Detekcia zrážok	59
6.1.3	Detekcia prítomnosti hmly	59
6.2	Realizácia algoritmov	60
6.2.1	Segmentácia vozovky a analýza jej stavu	60
6.2.2	Detekcia zrážok	72
6.2.3	Detekcia hmly	75
6.3	Zhrnutie klasických metód	83
6.3.1	Obmedzujúce podmienky analýzy stavu vozovky	83
6.3.2	Obmedzujúce podmienky detekcie zrážok	85
6.3.3	Obmedzujúce podmienky detekcie hmly	86
6.4	Testovanie algoritmov klasických metód	87
6.4.1	Testovanie stavu vozovky	88
6.4.2	Testovanie prítomnosti hmly	88
6.4.3	Testovanie detekcie zrážok	88
7	Použitie neurónových sietí	90
7.1	Dataset a príprava dát	90
7.2	Výber architektúry	91
7.3	Výber architektúry	91
7.4	Dosiahnuté výsledky	92

8 Porovnanie klasických metód a neurónových sietí	98
8.1 Logika vyhodnocovania	98
Závěr	100
Literatúra	102
Zoznam príloh	108
A Výsledné grafy použitých modelov	109

Zoznam obrázkov

1.1	Pyrometer pre bezkontaktné meranie teploty vozovky [7]	17
1.2	NTC termistor pre kontaktné meranie teploty vozovky [8]	17
1.3	2D ultrazvukový anemometer [9]	19
1.4	Snímač teploty, relatívnej vlhkosti a tlaku [10]	20
1.5	Automatický váhový zrážkomer [11]	21
1.6	Zrážkomer fungujúci na princípe Dopplerovho radaru [12]	22
1.7	Transmisometer [13]	23
1.8	Ceilometer [1]	24
1.9	Príklad dopravnej meteostanice [14]	26
2.1	Štyri možné úrovne reprezentácie obrázku [19]	28
2.2	Digitálny obrázok ako matica $M \times N$	29
2.3	Mapovanie bodov kartézianskej roviny do Houghovho priestoru [26]	33
2.4	Alternatívna reprezentácia priamky a jej mapovaniu do Houghovho priestoru [26]	34
2.5	Vstupný obrázok, štruktúrny element a výsledok erózie [28]	35
2.6	Výsledok uzavretia [29]	35
2.7	Vstupný obrázok, štruktúrny element a výsledok dilatácie [28]	36
2.8	Výsledok otvorenia [29]	36
2.9	Obrázok v časovej oblasti a jeho transformácia do frekvenčnej oblasti [30]	37
3.1	Vrstvy ANN [29]	38
3.2	Priebeh aktivačných funkcií a ich derivácií	41
3.3	Architektúra CNN	43
3.4	Ukážka 2D konvolúcie [34]	44
3.5	Ukážka funkcie max-pooling [34]	44
3.6	Ukážka architektúry VGG16 [36]	45
3.7	RNN [35]	46
3.8	Reziduálny blok [37]	47
3.9	ResNet18 [37]	47
4.1	Postup algoritmu [38]	50
4.2	Efekt hmly na hrany [40]	51
4.3	Nájdenie inflexného bodu [40]	52
4.4	Dosiahnuté výsledky CNN-RNN [48]	54
5.1	Ukážka datasetu stavu vozovky	55
5.2	Ukážka datasetu počasia s obecnou scénou [16]	56
5.3	Ukážka datasetu počasia s dopravnou scénou [18]	57
6.1	Ukážka porovnania scény s hmlou a bez hmly [51] a ukážka sneženia	60

6.2	Postup algoritmu segmentácie založenej na hľadaní priamok	61
6.3	Ukážka metódy segmentácie s použitím Houghovej transformácie . . .	62
6.4	Ukážka č.2 metódy segmentácie s použitím Houghovej transformácie .	62
6.5	Postup algoritmu segmentácie založenej na Cannyho detektore	63
6.6	Ukážka metódy segmentácie s použitím Canny detektoru	65
6.7	HSV farebný model [52]	67
6.8	Postup algoritmu segmentácie založenej na HSV modely	67
6.9	Porovnanie výsledkou segmentácie vozovky použitím všetkých kanálov HSV s ich samotným použitím	69
6.10	Postup algoritmu analýzy vozovky	70
6.11	Porovnanie histogramov z celého obrázku na suchú, zasneženú a mokrú vozovku. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov.	70
6.12	Porovnanie histogramov z vysegmentovanej oblasti suchej, zasneženej a mokrej vozovky. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov.	71
6.13	Postup algoritmu hľadania blobov	73
6.14	Výsledok použitia SimpleBlobDetector	73
6.15	Postup algoritmu detekcie zrážok	74
6.16	Výsledok metódy založenej na dynamickom modeli prostredia	75
6.17	Porovnanie histogramov slnečného, zamračeného a hmlistého počasia. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov [51].	76
6.18	Porovnanie histogramov slnečného, zamračeného a hmlistého počasia. Histogramy sú vykreslené zo šedotónového variantu hornej polovice zobrazených obrázkov.	77
6.19	Postup algoritmu detekcie hmly počítaním hrán	78
6.20	Porovnanie hrán slnečného (35951 resp. 3075), zamračeného (34822 resp. 4006) a hmlistého (22158 resp. 983) počasia	78
6.21	Postup algoritmu analýzy kontrastu	79
6.22	Porovnanie gradientu slnečného (19,02 resp. 42,3), zamračeného (29,54 resp. 68,38) a hmlistého (19,57 resp. 21,37) počasia použitím Sobelovho operátoru	80
6.23	Postup algoritmu analýzy obrazu vo frekvenčnej oblasti	81
6.24	Porovnanie magnitúdy frekvenčného spektra slnečného, zamračeného a hmlistého počasia	82
6.25	Ukážka parametrizácie viditeľnosti a intenzity zrážok	84
7.1	Grafický výstup natrénovaného modelu na celom datasete	94
7.2	Grafický výstup presností natrénovaného modelu na celom datasete .	94

7.3	Grafický výstup strát natrénovaného modelu na celom datasete pomocou confusion matice	95
7.4	Grafický výstup odhadu pravdepodobnosti modelom	97
A.1	Klasifikátor ResNet50	109
A.2	Klasifikátor ResNet50	110
A.3	Klasifikátor + 3 vrstvy ResNet50	110
A.4	Klasifikátor + 3 vrstvy ResNet50	111
A.5	Klasifikátor VGG16	111
A.6	Klasifikátor VGG16	112

Zoznam tabuliek

1.1	Vplyvy elementov počasia na dopravu	25
1.2	Štatistika nehôd motorových vozidiel v ČR v rokoch 2015-2019 [15] .	26
4.1	Zhrnutie výhod a nevýhod modelov	49
6.1	Zhrnutie charakteristických príznakov pre jednotlivé metódy	83
6.2	Výsledky algoritmov analýzy vozovky na troch stanoviskách, hodnoty sú uvedené v %	88
6.3	Výsledky algoritmov detekcie hmly na troch stanoviskách, hodnoty sú uvedené v %	89
6.4	Výsledky algoritmov detekcie zrážok na troch stanoviskách, hodnoty sú uvedené v %	89
7.1	Podoba použitého datasetu	91
7.2	Dosiahnuté výsledky použitím neurónových sietí	93
7.3	Štatistika finálneho modelu	95

Úvod

Predpoveď počasia patrí stále aj napriek výraznému pokroku vo výpočetnej technike medzi zložité a komplexné úlohy. Počasie hrá obrovskú úlohu v doprave. Doprava je nevyhnutná súčasť každého z nás, cestovanie do školy a práce, cestovanie za rodinou, transport tovaru, obchodné stretnutia a podobne. Výsledkom náhlych zmien počasia, pod ktoré sa podpisuje zmena klímy vplyvom globálneho otepľovania, sú čoraz častejšie prívalové dažde spôsobujúce povodne, čiže aj zaplavenie vozovky, minimálnu viditeľnosť, zatarasené cesty popadanými stromami po víchrici a podobne. Tieto prírodné udalosti výrazne ovplyvňujú bezpečnosť a rýchlosť dopravy, ba dokonca aj jej úplne dočasné zastavenie.

Hlavnou motiváciou práce je, že meteorologické stanice sú pomerne drahé a dnes sú čoraz častejšie inštalované kamery pre sledovanie dopravnej situácie, ktoré môžu byť použité aj pre odhad počasia.

Cieľom diplomovej práce je navrhnutie vhodných algoritmov, ktoré budú monitorovať stav vozovky, teda či je na ceste voda, snehová pokrývka alebo je vozovka suchá, a zároveň detekovať hmlu a prítomnosť zrážok a parametrizovať viditeľnosť a intenzitu zrážok. Vstupom do algoritmov je sekvencia obrázkov z dopravy, na ktoré budú uplatnené vhodné operácie počítačového videnia a výstupom bude určený stav počasia a vozovky. Uplatnenie algoritmov môže byť napríklad pri optimalizácii a údržbe vozovky v zime - kedy a ako často majú mať cestári výjazd, čo a koľko treba posypať a podobne. Druhým príkladom využitia môžu byť bezpečnostné a výstražné systémy ("Voda na ceste!", "Nebezpečenstvo šmyku!", "Pozor, námraza!", "Pozor, znížená viditeľnosť!").

Štruktúra práce je rozdelená následovne. Prvá kapitola obsahuje všeobecný teoretický úvod, súvislosť medzi dopravou a počasím, prečo je dobré počasie merať, kde je možné využitie, aké veličiny merať a aké použiť metódy.

V druhej kapitole sú teoreticky popísané základné stavebné bloky klasických metód počítačového videnia ako prahovanie alebo hranové operátory. V druhej polovici kapitoly je popísaný moderný prístup riešenia problémov počítačového videnia, a to neurónové siete.

Tretia kapitola popisuje existujúce riešenia analýzy počasia z kamerových obrázkov. V tejto kapitole sú popísané rôzne prístupy, podľa ktorých sa dajú detekovať rôzne stavy počasia.

Štvrtá kapitola je venovaná zberu a popisu vstupných dát, ktoré tvoria dokopy tri datasety.

Piata kapitola obsahuje na základe získaného prehľadu rozbor a návrh prístupov, ktoré sú použité na analýzu počasia. Sú v nej popísané tri metódy segmentácie vozovky, dve metódy analýzy vozovky, 4 metódy detekcie hmly a dve metódy detekcie

zrážok. Následná časť kapitoly sa venuje návrhu modelu neurónovej siete. Vyskúšal som hlavne architektúru ResNet a konvolúčnu neurónovú sieť VGG.

1 Úvod do problematiky

1.1 Počasie

Pojem počasie popisuje dočasné podmienky atmosféry, vrstvy vzduchu, ktorá obklopuje planétu Zem. Stav atmosféry je závislý na konkrétnom mieste a čase na našej planéte. Stále sa však pohybuje a mení zo dňa na deň, z hodiny na hodinu, v extrémnych podmienkach aj z minúty na minútu.

Existuje sedem základných elementov, ktoré spolu vytvárajú počasie. Sú to teplota, atmosférický tlak, vietor, vlhkosť, zrážky, viditeľnosť a oblačnosť. Jednotlivé zmeny základných komponentov umožňujú predpoveď počasie v blízkej budúcnosti. Pre potreby dopravy však okrem stavu okolia (teplota vzduchu, rýchlosť a smer vetra) musíme brať do úvahy aj stav vozovky - či je suchá alebo mokrá, aká vrstva snehu ju pokrýva, jej teplota pre možnosť predvídania tvorby poľadovice a podobne.

1.1.1 Teplota a jej meranie

Teplota je meranie množstva kinetickej energie prítomnej vo vzduchu, ktoré sa fyzicky prejavuje vo forme tepla alebo chladu. Čím sa molekuly a atómy vzduchu pohybujú rýchlejšie, tým je teplota v atmosfére vyššia a opačne, čím sa molekuly a atómy vo vzduchu pohybujú pomalšie, tým menej energie resp. tepla vygenerujú. Jej zmeny môžu byť ovplyvnené napr. vetrom a slnečným žiarením.

Pre meteorologické účely sa meria teplota rôznych médií. Najčastejšie je meraná teplota vzduchu atmosféry v rôznych nadmorských výškach. Ďalšie príklady merania môžu byť teplota povrchu zeme, teplota v rôznych hĺbkach pôdy a teplota v rôznych hĺbkach jazier, morí a oceánov. Tieto merania sú potrebné spoločne, alebo nezávisle, lokálne alebo globálne ako vstup do numerických modelov predpovede počasia. Teplota sa meria teplomerom najčastejšie v jednotkách stupňov Celzia [2, 3, 4].

Pre uplatnenie v doprave sa merajú hlavne dve teploty, teplota vozovky a teplota vzduchu. Na zisťovanie teploty vozovky sa používajú dva typy snímačov. **Bezkontaktné snímače** su inštalované na stĺpoch alebo mostíkoch vedľa vozovky. Tento typ snímačov je založený na pyrometrii. Základným princípom pyrometra je, že meria teplotu objektu snímaním množstva, koľko tepla/žiarenia tento objekt emituje. Na základe detekovanej intenzity žiarenia sa určuje teplota.

Kontaktné snímače teploty vozovky sú umiestnené v strede jazdného pruhu alebo na okraji vozovky. Používa sa NTC termistor. Termistory sú teplotne citlivé polovodiče, ktoré vykazujú pri zmene teploty zmenu odporu kvôli rôznej koncentrácii voľných nosičov náboja. NTC termistory majú negatívny teplotný koeficient, čo znamená, že s rastúcou teplotou odpor klesá.



Obr. 1.1: Pyrometer pre bezkontaktné meranie teploty vozovky [7]



Obr. 1.2: NTC termistor pre kontaktné meranie teploty vozovky [8]

Medzi často používané snímače teploty vzduchu patrí **odporový kovový snímač** alebo už spomenutý **odporový polovodičový snímač (termistor)**. Odporový kovový snímač pre meranie teploty používa meranie elektrického odporu materiálu, ktorého odpor sa známym spôsobom mení s teplotou materiálu [1].

1.1.2 Atmosférický tlak a jeho meranie

Atmosférický tlak možno definovať ako hmotnosť atmosféry. Hmotnosť všetkých atómov a molekúl vzduchu vytvára tlak vplyvom gravitačnej sily Zeme. Najvyšší tlak je blízko povrchu zeme lebo je tu najväčšie množstvo vzduchu a s rastúcou nadmorskou výškou je vzduchu čoraz menej, a tým tlak klesá. Zmeny atmosférického tlaku prinášajú zmeny počasia. Systém vysokého tlaku má vyšší tlak v strede a zvyčajne spôsobí ochladenie a jasnú oblohu. Systém nízkeho tlaku má nižší tlak vo svojom strede ako v oblastiach okolo neho. Môže priniesť zrážky a oteplenie. Okrem skutočného tlaku je potrebné určiť jeho trend. Trend tlaku je charakter a veľkosť zmeny

atmosférického tlaku po dobu pozorovania. Trend tlaku sa skladá z dvoch častí, a to zmeny tlaku a tlakovej charakteristiky. Zmena tlaku je rozdiel medzi nameranými hodnotami tlaku na začiatku a na konci merania. Tlaková charakteristika popisuje ako sa tlak za dané meranie zmenil, či klesá alebo rastie [2, 3, 4].

Meria sa v jednotkách hPa ($100 \text{ Pa} = 1 \text{ hPa}$) pomocou tlakomera. Používané typy v meteorológii sú **kapacitný tlakomer** alebo **piezoelektrický tlakomer**.

Kapacitný snímač tlaku pracuje na princípe, že ak dôjde k deformácii snímačovej membrány medzi doskami kondenzátora rozdielovým tlakom, dôjde k nerovnováhe kapacity medzi ním a dvomi doskami. Jedna elektróda je tvorená membránou citlivou na tlak a druhá je fixná.

Piezoelektrický snímač tlaku je digitálny tlakomer využívajúci vlastnosti piezoelektrických materiálov, ako je kremeň, na generovanie náboja na povrchu, keď sa na neho aplikuje tlak. Veľkosť náboja je úmerná použitej sile a polarita vyjadruje jeho smer. Náboj sa hromadí a rýchlo sa rozptýli pri zmene tlaku, čo umožňuje meranie rýchlo sa meniacich dynamických tlakov [1].

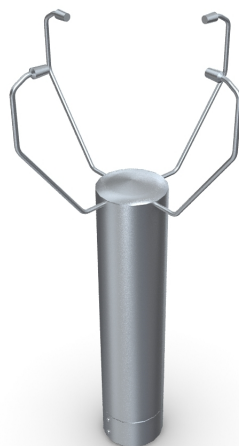
1.1.3 Vietor a jeho meranie

Vietor je pohyb vzduchu z oblasti vysokého tlaku do oblasti nízkeho tlaku v atmosfére. Rýchlosť a sila vetra závisí na vzájomnej vzdialenosti a veľkosti rozdielu tlaku vzduchu týchto dvoch oblastí. Čím sú oblasti bližšie a majú väčší rozdiel tlaku, tým je vietor silnejší a rýchlejší. Avšak kvôli tomu, že sa planéta otáča okolo svojej osi, nie je trajektória vzduchu priama, ale vychýlená. Za zakrivenie trajektórie môže byť jav, ktorý sa nazýva Coriolisov efekt. Pozorovania alebo merania vetra sú potrebné na monitorovanie a predpovedanie počasia, na pravdepodobnosť poškodenia vetrom a odhad veternej energie. Mnoho aplikácií vyžaduje informácie o nárazoch vetra. Takéto aplikácie poskytujú predpoveď na krátke časové obdobie pre vzlety a pristávanie lietadiel [2, 3, 4, 5, 6].

Pre meranie vetra sa zameriavame na spriemerovaný horizontálny vietor, ktorý sa zvyčajne vyjadruje v polárnych súradniciach ako rýchlosť a smer. Rýchlosť vetra je meraná v jednotkách m s^{-1} (meter za sekundu) a smer v jednotkách $^{\circ}$ (stupeň). Rýchlosť povrchového vetra sa zvyčajne meria **anemometrom**.

Ultrazvukový anemometer používa ultrazvukové vlny na stanovenie okamžitej rýchlosti vetra meraním toho, ako sa zvýši alebo zníži doba prechodu ultrazvukového impulzu medzi dvojicou vysielača a prijímača. Pre meranie zároveň aj smeru vetra sa používa 2D ultrazvukový anemometer.

Miskové anemometre sa používajú na stanovenie rýchlosti vetra a skladajú sa z rotora a generátora signálu. V dobre navrhnutých systémoch je uhlová rýchlosť



Obr. 1.3: 2D ultrazvukový anemometer [9]

misky priamo úmerná rýchlosti vetra. Prístroj sa skladá z troch alebo štyroch misiek, ktoré sú spojené s vertikálnym hriadeľom na otáčanie. Aspoň jedna miska je vždy oproti fúkajúcemu vetru. Rýchlosť vetra je prevádzaná na rotačný krútiaci moment. Prevodník v anemometri prevádza tento rotačný pohyb na elektrický signál, z ktorého je vypočítaná skutočná rýchlosť vetra [1].

1.1.4 Vlhkosť a jej meranie

Vlhkosť vyjadruje množstvo vody (vodnej pary) v atmosfére v daný čas. Vzduch dokáže zadržiavať vodu v plynovom skupenstve. Množstvo zadržanej vodnej pary závisí od teploty a tlaku vzduchu. Teplý vzduch môže obsahovať viac pary ako studený. Pri konkrétnej teplote existuje určité maximálne množstvo vody, ktoré je možné zadržať vzduchom. To predstavuje nasýtený stav alebo 100% vlhkosť. Relatívna vlhkosť vzduchu je podiel vodnej pary skutočnej vo vzduchu pri danej teplote v porovnaní s maximálnym možným množstvom pri tejto teplote. Môže sa líšiť od takmer nulovej nad púšťou až po 100 % v hustej hmle alebo daždi. Ďalším veľmi užitočným meraním vlhkosti je rosný bod - teplota, pri ktorej by relatívna vlhkosť dosiahla 100% pri súčasnom množstve prítomnej vodnej pary. Vyššie rosné body zodpovedajú väčšiemu množstvu vlhkosti. Relatívna vlhkosť sa meria vlhkomerom.

Kapacitný snímač relatívnej vlhkosti využíva zmenu elektrických vlastností materiálu pri prijímaní premenlivého množstva vodnej pary zo vzduchu. Aktívna časť snímača vlhkosti pozostáva z polymérového filmu vloženého medzi dve elek-



Obr. 1.4: Snímač teploty, relatívnej vlhkosti a tlaku [10]

tródy, aby vytvoril kondenzátor. Horná elektróda je priepustná pre molekuly vody a polymér (veľká molekula tvorená opakujúcimi sa menšími jednotkami) úmerne absorbuje vodu s narastajúcou relatívnou vlhkosťou.

Pozorovania teploty sú nevyhnutné popri pozorovaní vlhkosti, pretože hodnoty teploty sa používajú na výpočet ďalších veličín vlhkosti (napríklad rosného bodu) z relatívnej vlhkosti. Z toho dôvodu je senzor vlhkosti zvyčajne umiestnený v sonde spolu s kompaktným snímačom teploty [1].

1.1.5 Zrážky a ich meranie

Pod pojmom zrážky si môžeme predstaviť vodu v rôznych skupenstvách, ktorá sa po kondenzácii zmení z vodnej pary na kvapalnú, prípadne ďalej na pevnú formu, ktorá padá na zem vplyvom gravitácie potom, ako je príliš ťažká na to, aby zostala v oblaku. Keď Slnko ohrieva povrch Zeme, nad pevninou alebo vodou, ohrieva aj vzduch priamo nad ním. Pretože teplý vzduch je ľahší ako studený vzduch, začne stúpať do atmosféry. Teplo tiež spôsobuje odparovanie povrchovej vody a zvyšuje sa množstvo vodnej pary vo vzduchu. Ako som už spomenul, teplota vzduchu klesá s nadmorskou výškou, stúpajúci teplý vzduch sa začína ochladzovať. Akonáhle sa ochladí na hraničnú teplotu, pri ktorej voda už nemôže zostať v plynnej forme, dôjde ku kondenzácii a vytvoreniu malých vodných kvapiek. Výsledkom sú zrážky, ktorých množstvo, skupenstvo a ďalšie vlastnosti závisia od aktuálnych atmosférických podmienok. Medzi najznámejšie formy zrážok patrí dážď, mrznúci dážď, sneh

a krupobitie.

Celkové množstvo zrážok, ktoré dopadnú na zem za určité obdobie, sa vyjadruje ako vertikálna hĺbka vody (alebo ekvivalent vody v prípade pevných foriem). Pre kvapalné zrážky sa používajú jednotky mm (milimeter) vyjadrujúce pomer objemu a plochy alebo v jednotkách kg m^{-2} (pomer hmotnosť na plochu). Jeden milimeter dažďa odpovedá jednému litru na meter štvorcový vody na povrchu, čo je približne desať milimetrov snehu, $1 \text{ mm} = 1 \text{ l m}^{-2}$. Meranie snehovej prikrývky sa meria v cm (centimeter). Rovnako ako pri teplote, aj pri zrážkach je potrebné merať nie len zrážky v atmosfére ale aj zrážky na vozovke - vrstva vody/snehu na vozovke a stav povrchu vozovky (suchá, mokrá, zamrznutá). Pre detekciu zrážok v atmosfére možno použiť váhový zrážkometer alebo Dopplerov radar. V prípade merania stavu vozovky sa používa optický princíp.

Automatický váhový zrážkometer nepretržite zaznamenáva hmotnosť nádoby, do ktorej padajú zrážky. Hmotnosť je vyhodnocovaná pomocou pružinového mechanizmu, alebo pomocou systému vyvažovacích závaží. Existuje mnoho ďalších zrážkometerov, napríklad akustický, optický alebo zrážkometer typu plovák.



Obr. 1.5: Automatický váhový zrážkometer [11]

Dopplerov radar s frekvenciou 24 GHz meria rýchlosť poklesu jednotlivých kvapky dažďa alebo snehu. Množstvo a intenzita zrážok sa vypočítajú z korelácie medzi veľkosťou kvapky a rýchlosťou padania. Dokáže merať typ zrážok, intenzitu zrážok a množstvo zrážok. Umožňuje rozoznať dážď od snehu [1].

Kombináciou meraní je možné detekovať suchý alebo mokrý sneh a rôzne formy ľadu. Jeden snímač využívajúci dve elektródy zabudované v ceste **meria elektrickú vodivosť povrchu** a ľahko rozlišuje medzi suchým a mokrým povrchom. Druhé meranie **iónovej polarizovateľnosti** určuje schopnosť povrchu udržať elektrický



Obr. 1.6: Zrážkomer fungujúci na princípe Dopplerovho radaru [12]

náboj. Medzi dvojicou elektród prebehne na krátky čas malý náboj a tie isté elektródy merajú zvyškový náboj, ktorý je vyšší, ak je prítomný elektrolyt s voľnými iónmi, napríklad slaná voda. Merania polarizovateľnosti a vodivosti môžu spolu rozlišovať medzi suchým, vlhkým a mokrým povrchom, mrazom, snehom, bielym ľadom a niektorými rozmrazovacími chemikáliami.

V súčasnosti sa čoraz častejšie používa asfalt s takzvanými otvorenými pórmí, ktorý má menšiu hlučnosť a dokáže lepšie odvádzať vodu ako bežný asfalt. Tento druh asfaltu sa však nedá rezať pri inštalácii zabudovaného cestného snímača, preto sa zvyšuje využitie bezkontaktných snímačov. Tieto snímače sú založené na metóde optickej infračervenej spektrometrie. Spektroskopický snímač vyžaruje lúč infračerveného svetla, ktorý dopadá na rozšírenú oblasť. Svetlo je v tejto oblasti čiastočne absorbované ľadom alebo vodou a sníma sa svetlo odrazené späť do detektora. Prítomnosť alebo neprítomnosť ľadu alebo vody sa potom počíta pomocou optických vlastností spätne odrážaného infračerveného svetla [1].

1.1.6 Viditeľnosť a jej meranie

Viditeľnosť predstavuje pozorovanie voľným okom a označuje vzdialenosť, do ktorej je vizuálne vnímanie predmetov počas dňa a svetiel v noci obmedzené atmosférickými podmienkami.

Počas dňa, čím je tmavý objekt vzdialenejší od pozorovateľa, tým sa javí čoraz viac ako svetlejší objekt. Vo veľkých vzdialenostiach sa stane nerozoznateľným od horizontu oblohy. Táto limitujúca vzdialenosť je **denná viditeľnosť**, ktorá závisí na prahu vizuálneho kontrastu oka pozorovateľa. Zvyšovanie jasnosti objektu so zvyšovaním vzdialenosti od pozorovateľa je spôsobená rozptylom slnečného žiarenia vzduchom a k difúznemu odrazu slnečného svetla od terénu. Väčšina prípadov zníženej a zlej viditeľnosti je spôsobená rozptylom a útlmom svetla väčšími časticami



Obr. 1.7: Transmisometer [13]

v zrážkach a hmle. Meria sa v jednotkách m (meter) alebo km (kilometer).

Viditeľnosť v noci sa zvyčajne berie ako maximálna vzdialenosť, v ktorej sú viditeľné nezaostrené svetlá. Svetelný zdroj bude viditeľný, pokiaľ osvetlenie presahuje prahovú hodnotu pre ľudské oko. Nočná viditeľnosť teda závisí od intenzity zdroja svetla, osvetlenia pozadia a prahu osvetlenia pozorovateľa.

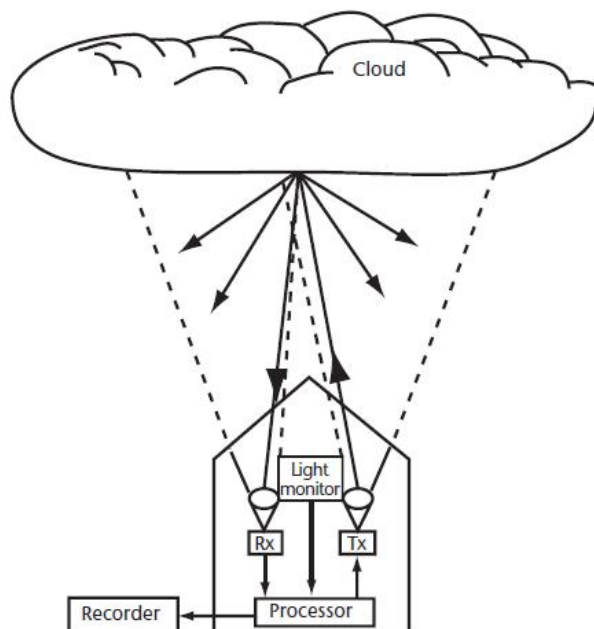
Absorbčný koeficient opisuje útlm intenzity svetla prechádzajúceho materiálom, čím je vyšší, tým menšia vzdialenosť preniknutia svetla do materiálu pred jeho absorbovaním.

Použitie **transmisometra** je metóda, ktorá sa najčastejšie používa na meranie absorbného koeficientu vo vodorovnom valci vzduchu medzi vysielateľom, ktorý poskytuje zdroj svetla a prijímačom obsahujúcim fotodetektor (obvykle fotodiódu) [1].

1.1.7 Oblačnosť a jej meranie

Opačným elementom ako je vlhkosť je oblačnosť, ktorá nemôže byť cítená ale je viditeľná. Oblaky sú jednou z najrýchlejších metód pre určenie aktuálneho a budúceho počasia a signalizujú prichádzajúcu zmenu počasia. Stúpajúce mraky značia stále počasia, zahusťujúce sa a klesajúce mraky znamenajú zrážky. Sú to vodné kvapky alebo voda v iných skupenstvách, ako napríklad ľad alebo snehové kryštály, ktoré

sa vytvorili potom, čo sa vodná para v atmosfére ochladí pod rosný bod a kondenzuje na malé ale viditeľné kvapôčky alebo ľadové kryštály. Základňa mračna (výška oblaku) označuje hladinu, pri ktorej stúpajúci vzduch dosiahne svoj rosný bod, napríklad hmla, známy mrak so svojou základňou blízko zemského povrchu. Mraky môžu ovplyvňovať množstvo slnečného žiarenia dopadajúceho na povrch Zeme.



Obr. 1.8: Ceilometer [1]

Pri skúmaní oblakov je meraná **výška oblaku** v jednotkách m (meter). Pomocou **laserového ceilometra** sa výška oblaku určuje meraním času potrebného na to, aby impulz koherentného svetla prešiel z vysielачa na základňu oblaku a vrátil sa k prijímaču. Hlavná časť žiarenia je rozptýlená od oblaku smerom nahor, ale jej časť je rozptýlená smerom dole a je zameraná prijímačom. Jednotkou **množstva oblaku** je okta, čiže koľko osmín oblohy je zakrytých mrakmi. Existujú kamery špeciálne navrhnuté na meranie množstva oblakov. Pomocou zakrivených zrkadiel môžu snímať celkovú oblohu. Obrázok z oblohy sa analyzuje pomocou algoritmu, ktorý určuje, či je v každom pixeli prítomný mrak pomocou nameranej farby. Suma všetkých pixelov dáva výsledok množstva mraku. Rozsiahly pokrok softvéru umožňuje pomocou analýzy obrázku určiť zároveň aj typ oblaku [1].

1.2 Počasie a doprava

Počasie výrazne ovplyvňuje prevádzku dopravných systémov, na ktoré sa všetci spoliehame - od automobilov a nákladných vozidiel spomalených mokrým povrchom vo-

zovky, cez lietadlá oneskorené silným vetrom až po vlaky odstavené ľadom a snehom. Každodenné prevádzkové rozhodnutia v dopravnom sektore, ako napríklad množstvo nákladu, ktoré lietadlo alebo nákladná loď dokáže bezpečne prepraviť na požadované miesto, musia brať do úvahy aktuálny stav počasia a ako sa tento stav bude vyvíjať. Cesty, diaľnice, mosty, železnice, letiská, prístavy a iné dopravné infraštruktúry sú optimalizované voči očakávanému rozsahu poveternostných podmienok. V prípade extrémnejších zmien, ktoré sú čoraz bežnejšie a ťažšie sa predpovedajú do vzdialenejšej budúcnosti, sa dopravné infraštruktúry stávajú nebezpečnejšími, nepredvídateľnejšími a menej spoľahlivejšími. Stúpajúce hladiny morí a oceánov, či intenzívne záplavy ovplyvňujú činnosť prístavov alebo mostov, vysoké teploty ohýbajú koľajnice a podobne.

Obrázok 1.9 znázorňuje modernú meteostanicu, ktorá sa používa pre monitorovanie poveternostných podmienok v doprave. Je to súbor rôznych typov snímačov, ktoré som spomenul v predchádzajúcej časti kapitoly. Tieto dopravné meteostanice sú montované na stĺpy na strategických miestach dopravných ťahoch. Dnes sú bežne doplnené aj o snímacie kamery.

Tab. 1.1: Vplyvy elementov počasia na dopravu

Parameter počasia	Kategória	Dôsledky
Zrážky a viditeľnosť	mrznúce zrážky, hromadenie snehu, tekuté zrážky, záplavy, hmla, opar, prach a slnečné žiarenie	znížená rýchlosť, meškanie, namáhanie pneumatík vozidla, slabšie brzdenie, uzávery ciest a zhoršená viditeľnosť
Teplota	teplota vzduchu a povrchu	namáhanie častí vozidla a dopravnej infraštruktúry, pri vysokých teplotách rýchlo sa kaziaci náklad, pokrivenie koľajníc, obmedzenie rýchlosti vlakov
Vietor	rýchlosť vetra	zhoršená viditeľnosť, nestabilita vozidla

Tabuľka 1.1 konkrétnejšie popisuje ako vplývajú vybrané elementy počasia na dopravu. Znížená viditeľnosť, zaplavená cesta, namáhanie komponentov vozidla a jeho nestabilita vedie k výraznému obmedzeniu bezpečného vedenia dopravného prostriedku. Toto riziko, bohužiaľ, pomerne často vedie k dopravnej nehode, ktorá vedie k majetkovým škodám a k stratám na ľudských životoch.



Obr. 1.9: Príklad dopravnej meteostanice [14]

Tab. 1.2: Štatistika nehôd motorových vozidiel v ČR v rokoch 2015-2019 [15]

Rok	Celkový počet nehôd motorových vozidiel	Počet nehôd vplyvom zhoršeného počasía	Počet obetí nehôd spolu	Počet obetí nehôd vplyvom zlého počasía
2015	78201	5881	601	61
2016	82981	6952	496	38
2017	86187	6921	461	54
2018	86632	5993	521	46
2019	86315	5803	508	44

Tabuľka 6.3 zobrazuje štatistiku dopravných nehôd v Českej republike za posledných päť rokov (2015-2019). Zo štatistiky je jasné, že dopravné nehody vplyvom počasia nemožno zanedbať. Z tohto dôvodu sa na strategických miestach čoraz čas-

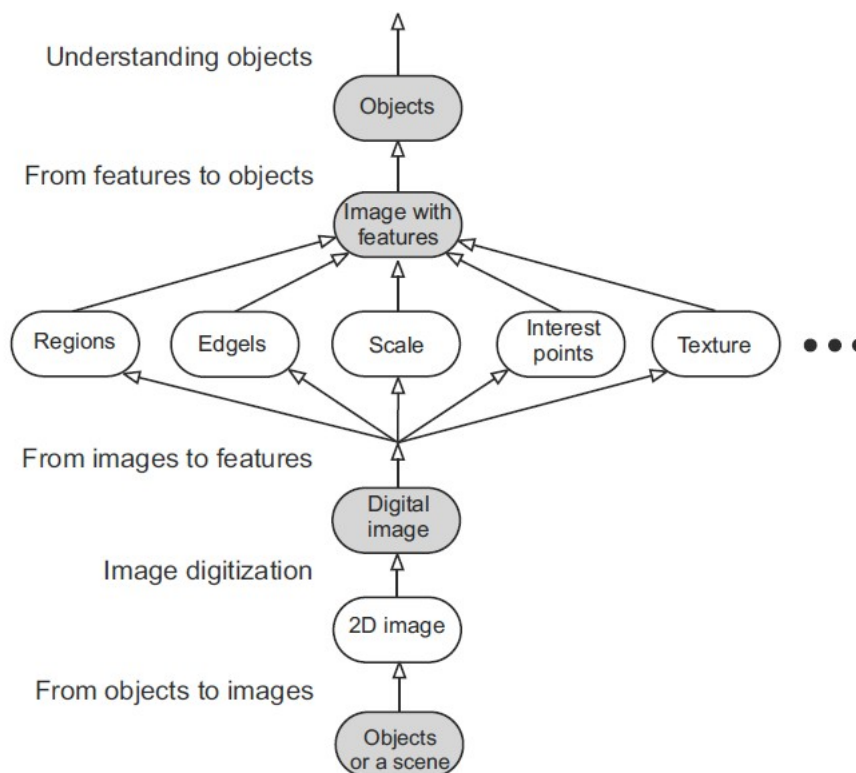
tejšie objavujú monitorovacie kamerové systémy, ktoré môžu byť doplnené senzormi a tak tvoriť meteostanicu. Meracia technika je natoľko vyspelá, že bežne používané snímače v dopravných meteostaniciach sú takzvané all-in-one snímače - integrované rôzne snímače, napríklad teplota vzduchu, teplota vozovky, stav vozovky, relatívna vlhkosť. Známymi spoločnosťami, ktoré sa venujú takýmto snímačom sú Vaisala alebo Lufft. Pomocou siete takýchto systémov sa môže zvýšiť efektivita, rýchlosť a hlavne bezpečnosť dopravy.

2 Spracovanie obrazu

Pochopenie obrazu počítačom možno považovať za pokus o nájdenie vzťahu medzi vstupnými obrazmi a predtým vytvorenými modelmi pozorovaného sveta [19]. Prechod od vstupného obrázka k modelu redukuje informácie obsiahnuté v obraze na relevantné informácie pre konkrétnu aplikáciu. Tento proces možno rozdeliť do niekoľkých krokov alebo zjednodušene na nízkoúrovňové a vysokoúrovňové metódy. Spodná vrstva obsahuje nespracované obrazové informácie, ktoré interpretujú vyššie úrovne.

Nízkoúrovňové spracovania zvyčajne využívajú len veľmi málo poznatkov o obsahu obrázkov. Ak počítač pozná obsah obrázku, pravdepodobne ho poskytol nejaký vysokoúrovňový algoritmus.

Vysokoúrovňové spracovanie sa snaží v čo najväčšej miere napodobniť ľudské videnie a schopnosť rozhodovať sa podľa informácií obsiahnutých v obraze [19].



Obr. 2.1: Štyri možné úrovne reprezentácie obrázku [19]

Ak sa spoja tieto dve úrovne, dostaneme určitú postupnosť krokov spracovania a to - **snímanie a digitalizácia, predspracovanie, segmentácia a popis**.

2.1 Digitalizácia

Aby počítač mohol pracovať s digitálnym obrázkom, musí byť reprezentovaný pomocou vhodnej diskretnej dátovej štruktúry, napríklad maticou. Obrázok je nasnímaný pomocou snímača, ktorý ho vyjadruje ako spojitú funkciu $f(x, y)$ dvoch súradníc v rovine. Digitalizácia obrazu teda znamená, že spojitá funkcia $f(x, y)$ je navzorkovaná do matice, ktorá má M riadkov a N stĺpcov. Bod matice sa nazýva pixel a nesie jasový údaj v danom mieste roviny [19].

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & f(0,2) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \dots & f(M-1,N-1) \end{bmatrix}$$

Obr. 2.2: Digitálny obrázok ako matica $M \times N$

Kvantizácia obrazu priradzuje každej spojitaj vzorke celočíselnú hodnotu, čím sa rozsah spojitaj obrazovej funkcie $f(x, y)$ rozdelí na K intervalov. Čím je vzorkovanie a kvantizácia jemnejšia (väčšie rozmery matice M a N a väčší počet intervalov K), tým je výsledná aproximácia spojitaj obrazovej funkcie presnejšia [21].

2.2 Predspracovanie

Predspracovanie popisuje operácie s obrázkom na najnižšej úrovni abstrakcie. Cieľom predbežného spracovania je preto vylepšenie obrazových údajov, ktoré potláča nežiaduce skreslenia, odstraňuje šum alebo vylepšuje niektoré obrazové prvky (detekcia hrán, zvýraznenie hrán, doostrenie obrazu) dôležité pre ďalšie spracovanie. Vstupný a výstupný obrázok predstavuje maticu hodnôt jasu. Predspracovanie nemení obsah obrazových informácií.

2.3 Segmentácia

Segmentácia obrazu je jedným z najdôležitejších krokov vedúcich k analýze spracovaných obrazových dát. Jej hlavným cieľom je oddelenie dôležitých objektov medzi sebou a od pozadia tak, aby sme dostali dôležité časti obrazu, ktoré korelujú s objektami alebo oblasťami reálneho sveta. Inými slovami, rozdelenie celého komplexného obrazu na menšie podskupiny pixelov nazývané obrazové objekty, ktoré môžu znížiť

zložitosť obrazu, a tak sa jeho analýza zjednoduší. Existuje viacero rôznych segmentačných metód [19, 21, 22, 23]. V nasledujúcich častiach kapitoly popíšem len tie metódy, ktoré majú blízko k mojej téme diplomovej práce.

2.3.1 Prahovanie

Prahovanie patrí k najstarším a najjednoduchším metódam segmentácie, ktorá sa neustále používa v jednoduchých aplikáciach. Je to rýchla a výpočtovo nenáročná metóda, jednoducho vykonateľná aj v reálnom čase. Mnoho objektov alebo oblastí obrazu sa vyznačuje konštantnou absorpciou svetla ich povrchov, na základe čoho je možné stanoviť absolútnu prahovú hodnotu na segmentovanie objektov a pozadia. Na najnižšej úrovni sa prahová hodnota T považuje za konštantnú. Tento prístup môže byť nepoužiteľný vzhľadom na množstvo šumu (nežiadúce informácie), ktoré obrázok obsahuje. Môžeme ho teda buď udržať konštantný, alebo ho dynamicky meniť na základe vlastností obrázka, čím získame lepšie výsledky. Ako som spomenul, výhodou prahovania je jeho jednoduchosť a naopak, nevýhodou je, že môžu byť vynechané mnohé detaily.

1. **Klasické prahovanie** pracuje s konštantnou hodnotou prahu a nahradí pôvodné hodnoty pixelov v obraze buď čiernou alebo bielou. Ak je hodnota pixelu (intenzita) menšia ako prahová hodnota, je nahradená čiernou (0), inak bielou (1). Výsledkom je binárny čierno-biely obraz.
2. **Otsu binarizácia** je založená na myšlienke, že ak má skúmaný obrázok vo svojom histograme práve dva vrcholy (bimodálny obrázok), možno ako prahovú hodnotu brať približne hodnotu v strede medzi vrcholmi. Inak povedané, automaticky vypočítava prahovú hodnotu z histogramu obrázka pre bimodálny obrázok.
3. **Adaptívne prahovanie** sa uplatňuje v aplikáciach, kde má obrázok odlišné podmienky pozadia a popredia v rôznych oblastiach. V takýchto typoch úlohy použitie konštantnej prahovej hodnoty nemá uspokojujúce výsledky. Adaptívny prístup umožňuje zmeniť prahovú hodnotu pre rôzne zložky obrazu. Algoritmus rozdelí obraz na rôzne menšie časti a vypočíta prahovú hodnotu pre tieto časti obrazu. Výsledkom sú rôzne prahové hodnoty pre rôzne oblasti [23].

2.3.2 Segmentácia založená na detekcii hrán

Zjednodušene povedané, je založená na princípe hľadania miest v obraze, kde dochádza k výrazným zmenám jasů. Detekcia hrán je proces lokalizácie hrán v obraze, čo patrí k dôležitým krokom pochopenia obrazu [19]. Dosahuje výborné výsledky pre obrázky s väčším kontrastom. Pre zašumené obrázky sa jej použitie neodporúča.

Hrana je vlastnosť spojená s jednotlivým pixelom a počíta sa z chovania obrazovej funkcie v susedstve tohto pixela. Je to vektorová premenná s dvoma zložkami, veľkosťou a smerom. Veľkosť hrany je veľkosť gradientu a smer hrany sa otáča vzhľadom na smer gradientu o -90° . Smer gradientu udáva smer maximálneho rastu funkcie. Veľkosť gradientu $|\text{grad}(x, y)|$ a smer gradientu Ψ sú spojité obrazové funkcie, ktoré možno vypočítat ako

$$|\text{grad}(x, y)| = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}, \quad (2.1)$$

$$\Psi = \arg\left(\frac{\partial g}{\partial x}, \frac{\partial g}{\partial y}\right), \quad (2.2)$$

kde $\arg(x, y)$ je uhol vzhľadom k ose x [19].

Algoritmy segmentácie založenej na detekcii hrán sú založené na rôznych diskontinuitách úrovne šedej, farieb, textúr, intenzity a podobne. Tieto algoritmy používajú hranové operátory, ktoré možno rozdeliť do dvoch skupín [19, 21].

1. Operátory aproximujúce **derivácie** obrazovej funkcie pomocou rozdielov

Tieto operátory nie sú invariantné voči otočeniu a preto je potrebné počítať v niektorých prípadoch odozvy s rôzne otočenými maskami [20].

Robertov operátor patrí k najstarším operátorom, je výpočtovo veľmi jednoduchý, používa masku 2×2 .

$$R_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, R_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Hlavnou nevýhodou tohto operátoru je vysoká citlivosť na šum, pretože na aproximáciu gradientu sa používa málo pixelov.

Prewittov operátor používa symetrické jadro, väčšinou s rozmerom 3×3 . Je definovaný jadrom v štyroch variantách otočenia (vodorovný smer, vertikálny smer a diagonály). Jednotlivé varianty sa môžu vzájomne kombinovať. Každý generuje kladnú alebo zápornú odozvu podľa toho, či je hrana v odpovedajúcom smere nábežná alebo zostupná. Informácia o smere sa využíva len obvykle, preto je možné zjednodušiť úlohu na kladné čísla. Ak je odozva výrazná, hrana je taktiež výrazná a opačne [20].

$$P_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, P_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix},$$

$$P_{xy} = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, P_{yx} = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}$$

Sobelov operátor je veľmi podobný s operátorom Prewitt a patria spolu medzi základné operátory detekcie hrán. Líšia sa vo váhach koeficientov.

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix},$$

$$S_{xy} = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}, S_{yx} = \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix}$$

- Operátory založené na **prechode nulou druhej derivácie** obrazovej funkcie
Princíp detekcie hrán založený na prechodoch nulou druhej derivácie skúma skutočnosť, že hrana zodpovedá náhlejšiemu zmene obrazovej funkcie. Prvá derivácia obrazovej funkcie by mala mať extrém v polohe hrany v obraze, a tak by druhá derivácia mala byť na rovnakej pozícii nulová. Druhá derivácia je výpočetnejšie jednoduchšia a má presnejšie výsledky. Operátory sú invariantné voči rotáciám [20] - jedna maska pre všetky smery.

Laplaciánsky operátor aproximuje druhú deriváciu a výsledkom je iba veľkosť hrany. Často používaná je maska s rozmermi 3 x 3 pre štvor-okolie resp. osem-okolie. Laplacián má rovnaké vlastnosti vo všetkých smeroch. Nevýhodou operátora je, že na niektoré hrany reaguje dvakrát.

$$L_4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}, L_8 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Cannyho detektor sa skladá z krokov filtrácie šumu, detekcie hrán (najčastejšie Sobel), ztenšenia nájdených hrán a prahovania s hystereziou. Hrúbka nájdených hrán nám neposkytuje žiadnu informáciu navyše, stačí nám len jej pozícia. V kroku ztenšenia hrán je porovnávanie jednotlivých pixelov obrázku s jeho susedmi naprieč hranou (napr. ak mám vertikálnu hranu, skúmam pixely o jednu pozíciu naľavo a o jednu napravo od skúmaného pixela). Na výstupe chceme mať obrázok s dominantnými hranami a preto použijeme hysterézne prahovanie (krok č. 4). Hysterézne prahovanie si možno predstaviť ako dvoj-úrovňové prahovanie (nižší a vyšší prah). Všetky hrany, ktoré sú pod nižším prahom sú automaticky ignorované a všetky hrany, ktoré sú nad vyšším prahom sú automaticky akceptované. Hrany medzi nižším a vyšším prahom sú zachované iba vtedy, ak sú spojené s hranou nad vyšším prahom [19].

2.3.3 Houghova transformácia

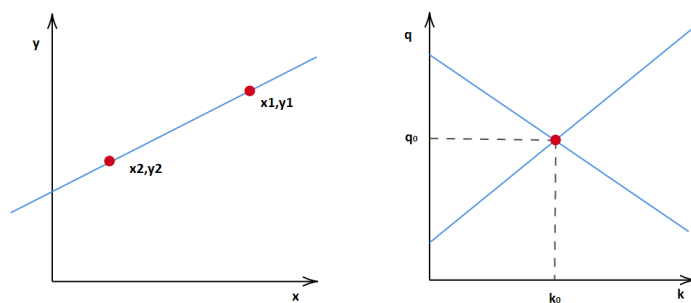
Houghova transformácia (HT) sa často používa v úlohach, kde je známy analytický popis hľadaného objektu - hľadanie priamiek, kružníc, trojuholníkov a pod. V prípade, kde nie je známy analytický tvar objektu, sa používa zobecnená Houghova transformácia [19, 26].

V karteziánskej rovine (os x a y) je priamka popísaná vzťahom

$$y = kx + q \quad (2.3)$$

, kde x a y sú súradnice bodu na tejto priamke, k je smernica priamky a q je priesečník s osou y . Priamka v karteziánskej rovine je vykresľovaná ako funkcia hodnôt x a y - vykreslenie priamky na základe počtu párov hodnôt x a y . Druhou možnosťou, ako vykresliť priamku je použiť hodnoty k a q . Toto sa využíva v Houghovom priestore (Hough Space) [26]. V prípade detekcie priamky, možno Houghov priestor zhrnúť ako:

1. Body v karteziánskej rovine sa mapujú ako priamky v Houghovom priestore
2. Priamky v karteziánskej rovine sa mapujú ako body v Houghovom priestore



Obr. 2.3: Mapovanie bodov karteziánskej roviny do Houghovho priestoru [26]

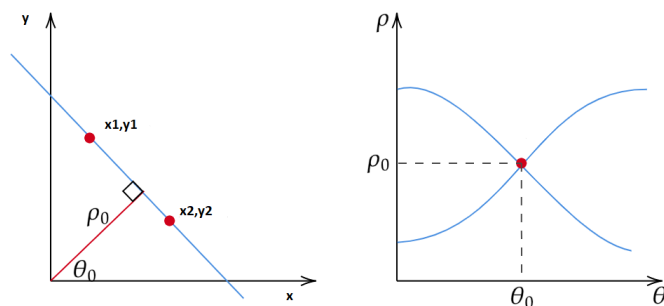
Pre nájdenie priamky v obraze sa vyhľadáva priesečník priamiek v Houghovom priestore. Ako efektívnejšia varianta sa javí rozdelenie obrázka na súbor malých rovnakých oblastí, kde je hľadaný maximálny počet priesečníkov priamiek [27].

Druhou možnosťou, ako vyjadriť priamku je **polárny súradnicový systém** [27]. V polárnom systéme je priamka popísaná vzťahom

$$P = x \cos \theta + y \sin \theta \quad (2.4)$$

, kde P je vzdialenosť od počiatku, θ je uhol od kladnej poloosi x k priamke, $x \cos \theta$ a $y \sin \theta$ sú vzdialenosti v smere x resp. smere y . Mapovanie bodov do Houghovho priestoru funguje rovnako s rozdielom, že sa generuje kosínusová krivka v Houghovom

priestore namiesto priamky. Táto varianta rieši problém nedefinovateľnosti smernice vertikálnych priamiek [26].



Obr. 2.4: Alternatívna reprezentácia priamky a jej mapovaniu do Houghovho priestoru [26]

2.4 Popis a klasifikácia

Ako predposlednou časťou reťazca spracovania obrazu je popis objektov. Tento krok patrí do vyššej úrovne spracovania obrazu. Cieľom popisu je získanie príznakov z vysegmentovaných dát. Príznaky musia vystihovať charakteristické vlastnosti objektu. Slúžia k následnej klasifikácii objektov. Hlavným cieľom klasifikácie obrazu je na základe získaných príznakov predpovedať konkrétnu triedu, do ktorej obrázok patrí.

2.5 Morfologické operácie

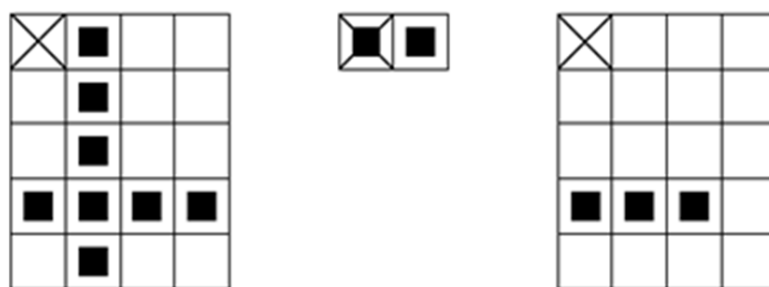
Morfologické operácie nie sú ďalším krokom v reťazci spracovania obrazu, ale majú tak širokú škálu použitia, že sa vyskytujú v predspracovaní obrazu (filtrácia šumu, zjednodušenie tvarov), v segmentácii objektov od pozadia, vo vylepšení štruktúry objektu (ztenšenie/zhrubnutie hrán) a nakoniec aj v kvantitatívnom popise objektov (plocha, priemer, projekcia).

Morfologické operácie sú jednoduché operácie založené na tvare obrazu. Zvyčajne sa vykonávajú na binárnych obrazoch. Potrebujú dva vstupy - pôvodný obrázok a štruktúrny element. Štruktúrny element alebo jadro určuje povahu operácie. Dve základné morfologické operácie sú erózia (erosion) a dilatácia (dilation) a ich variácie morfologické otvorenie (opening) a morfologické uzavretie (closing).

2.5.1 Erózia

Základná myšlienka erózie je zmenšenie objektu, ktorý je tvorený pixelmi s hodnotou 1. Ide o vektorové odčítanie. Štruktúrny element alebo jadro prechádza obrazom. Pixel v pôvodnom binárnom obrázku bude ponechaný s hodnotou 1, iba ak sú pod všetkými 1 v jadre aj 1 v originálnom obrázku, inak je daný pixel vynulovaný. Všetky pixely blízko hranice budú zlikvidované v závislosti od veľkosti jadra. Takže hrúbka alebo veľkosť objektu sa zmenšuje alebo inými slovami sa v obraze zmenšuje biela oblasť. Je to užitočné na odstránenie malých šumov, odpojenie dvoch pripojených objektov a podobne.

Dilatácia nasledovaná eróziou sa nazýva **uzavretie**. Uzavretie sa využíva v prípade vyplnenia malých čiernych oblastí v objekte [19, 28, 29].



Obr. 2.5: Vstupný obrázok, štruktúrny element a výsledok erózie [28]

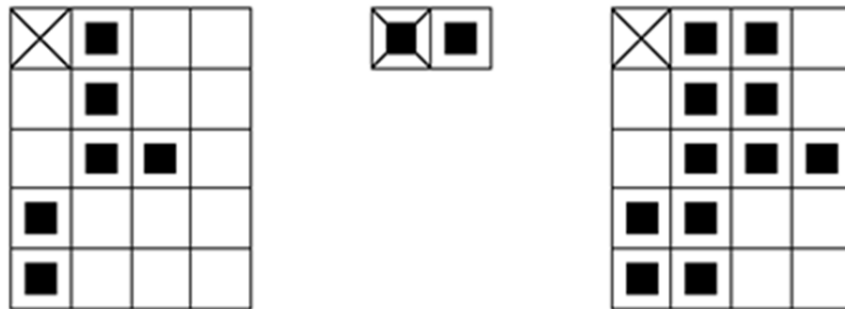


Obr. 2.6: Výsledok uzavretia [29]

2.5.2 Dilatácia

Dilatácia je duálna k erózii. Ide o vektorové sčítanie. Tu je pixel 1, ak aspoň jeden pixel pod jadrom je 1. Dilatácia zväčšuje bielu oblasť (tvorenú pixelmi s hodnotou 1) alebo vyplní čierne diery vnútri tejto oblasti. V prípadoch, ako je odstránenie šumu, eróziu zvyčajne nasleduje dilatácia - **otvorenie**. Erózia odstraňuje biely šum, čím zmenšuje aj objekt. Aby sa objekt znovu zväčšil, použije sa dilatácia - zbavím

sa šumu, ale veľkosť objektu je zachovaná. Dilatácia je tiež užitočná pri spájaní menších častí objektu.



Obr. 2.7: Vstupný obrázok, štruktúrny element a výsledok dilatácie [28]

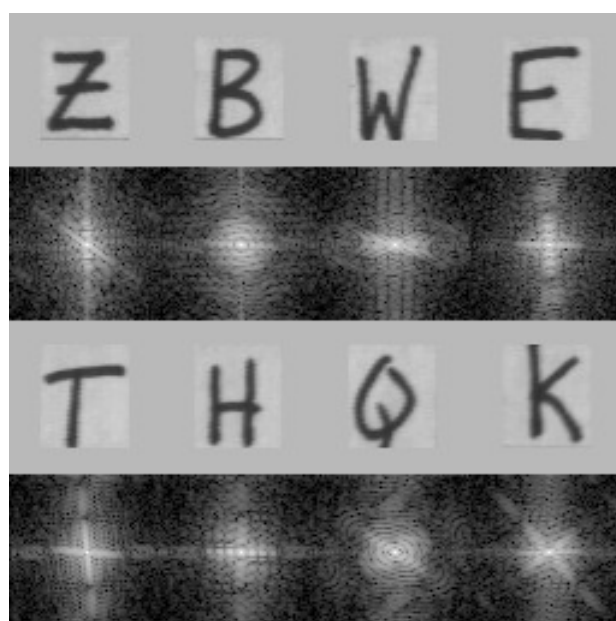


Obr. 2.8: Výsledok otvorenia [29]

2.6 Analýza obrazu vo frekvenčnej oblasti

2D Fourierova transformácia je transformácia, ktorá má na vstupe 2D dáta a na výstupe vráti iné 2D dáta. Inak povedané, prevedie dáta z časovej oblasti do oblasti frekvenčnej. Pri prevode nedochádza ku strate dát. Významné použitia Fourierovej transformácie sú kompresia obrazu, OCR (Optical Character Recognition), zaostrenie obrazu, detekcia hrán alebo filtrácia šumu.

Obrázok frekvenčnej domény nám hovorí, ako veľmi sú jednotlivé frekvencie v pôvodnom obrázku obsiahnuté. Pixely okolo stredu predstavujú zložky s nízkou frekvenciou a pixely na vonkajšej strane vyššie frekvenčné zložky. Ak je stred jasnejší, znamená to, že v pôvodnom obrázku prevládajú komponenty s nízkou frekvenciou.



Obr. 2.9: Obrázok v časovej oblasti a jeho transformácia do frekvenčnej oblasti [30]

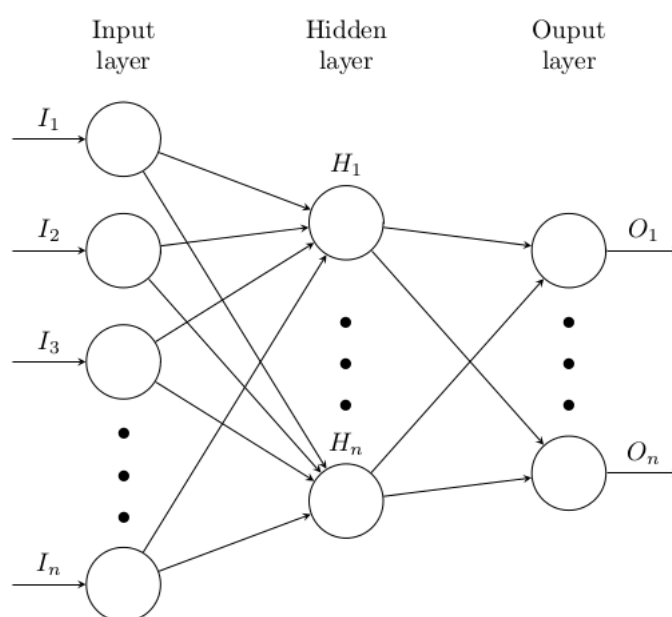
3 Neurónová sieť

Neurónové siete a hĺbkové učenie v súčasnosti poskytujú najlepšie riešenia mnohých problémov v oblasti počítačového videnia, rozpoznávania objektov alebo spracovania prirodzeného jazyka. Čoraz častejšie vychádzajú odborné články, ako sa umelá inteligencia naučila maľovať, skladať hudbu alebo vytvárať 3D modely. Pod pojmom umelá neurónová sieť (ANN) si možno predstaviť výpočtový model inšpirovaný spôsobom spracovania informácie biologickej neurónovej siete v ľudskom mozgu [31].

3.1 Vrstvy neurónovej siete

Základným výpočtovým blokom každej neurónovej siete je umelý neurón (artificial neuron). Neurónové siete sú tvorené vrstvami neurónov. Prvá je vstupná vrstva (input layer), posledná je výstupná vrstva (output layer) predpovedajúca finálny výstup. Medzi vstupnou a výstupnou vrstvou sa nachádzajú ukryté vrstvy (hidden layers), ktoré vykonávajú väčšinu výpočtov.

$$Y = \text{Activation} \left[\sum (\text{weight} \cdot \text{input}) + \text{bias} \right] \quad (3.1)$$



Obr. 3.1: Vrstvy ANN [29]

1. **Vstupná vrstva** dostáva vstupy a predáva informácie do následujúcej vrstvy. Vo vstupnej vrstve neprebiehajú žiadne výpočty.

2. **Skryté vrstvy** vykonávajú výpočty a následne prenášajú váhy zo vstupnej vrstvy do nasledujúcej skrytej alebo úplne poslednej výstupnej vrstvy. Neurónova sieť nemusí mať žiadnu skrytú vrstvu, alebo ich môže mať až niekoľko desiatok. Každá môže mať inú aktivačnú funkciu. Sieť s viacerými skrytými vrstvami sa označuje ako hlboká neurónová sieť (deep neural network).
3. **Výstupná vrstva** nesie použitím aktivačnej funkcie predpovedaný výstup v požadovanom formáte.

3.2 Aktivačné funkcie

Aktivačná funkcia je prahovacia funkcia, ktorej výstup je použitý ako vstup nasledujúcej vrstvy (forward propagation). Medzi základné a najpoužívanejšie aktivačné funkcie patrí lineárna funkcia, sigmoid, tanh, ReLU a softmax [33].

Lineárna funkcia

Lineárna funkcia je definovaná ako

$$f(x) = a \cdot x \quad (3.2)$$

, kde a je ľubovoľná konštanta. Aktivácia je úmerná vstupu. Gradient pri lineárnej funkcii nadobudne konštantnú hodnotu (rovnú konštantu a), ktorá nezávisí na vstupnej hodnote x . To znamená, že váhy a odchýlky sú v rámci spätného šírenia aktualizované stále s rovnakým faktorom aktualizácie. Z toho dôvodu je lineárna funkcia vhodná pre jednoduché úlohy.

Gradient hovorí o zmene všetkých váh s ohľadom na zmenu chyby. Možno si ho predstaviť ako sklon funkcie, čím je sklon väčší, tým je väčší gradient a tým rýchlejšie sa model učí.

Sigmoid

Je nelineárna aktivačná funkcia, ktorá transformuje vstupné hodnoty do intervalu $[0, 1]$. Preto sa často používa v úlohách, ktoré majú ako výstup predpovedanú pravdepodobnosť.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (3.3)$$

Z priebehu derivácie zobrazenej na obrázku 3.2 možno vidieť, že významné hodnoty gradientu sú v intervale $[-3, 3]$. V ostatných častiach sú hodnoty gradientov veľmi nízke, čo spôsobí postupné zanikanie gradientu (vanishing gradient problem). Nevýhodou sigmoidu je, že nie je centrováný okolo nuly, čo má za následok, že výstupy všetkých neurónov budú s rovnakým znamienkom.

Tanh

Hyperbolický tangens je veľmi podobný funkcii sigmoid len s tým rozdielom, že je symetrický voči počiatku a interval namapovaných hodnôt je $[-1, 1]$. Vstupy do nasledujúcich vrstiev teda nebudú mať vždy rovnaké znamienko.

$$\tanh(x) = \frac{2}{1 + e^{-2 \cdot x}} - 1 = 2 \cdot \text{sigmoid}(2 \cdot x) - 1 \quad (3.4)$$

Z priebehu derivácie, ktorá je zobrazená na obrázku 3.2 možno vidieť, že v porovnaní s deriváciou funkcie sigmoid má tanh strmší gradient. Kvôli vyššiemu gradientu a symetrii voči stredu, sa javí tanh ako lepšia voľba voči funkcii sigmoid.

ReLU (Rectified Linear Unit)

ReLU si získala popularitu tým, že neaktivuje všetky neuróny v rovnaký čas. Je výpočetne jednoduchšia ako sigmoid a tanh lebo pre negatívne hodnoty vstupu sa neuróny neaktivujú, čo urýchľuje proces učenia modelu.

Hlavnou výhodou ReLU je, že odstraňuje problém miznúceho gradientu. Väčšina dnešných hlbokých neurónových sietí používa ReLU v skrytých vrstvách.

$$f(x) = \max(0, x) \quad (3.5)$$

Z obrázka derivácie je vidieť, že na ľavej poloosi je gradient rovný 0 a preto sa váhy a odchýlky pre niektoré neuróny neaktualizujú a vznikajú takzvané mŕtve neuróny, ktoré nikdy nebudu aktivované.

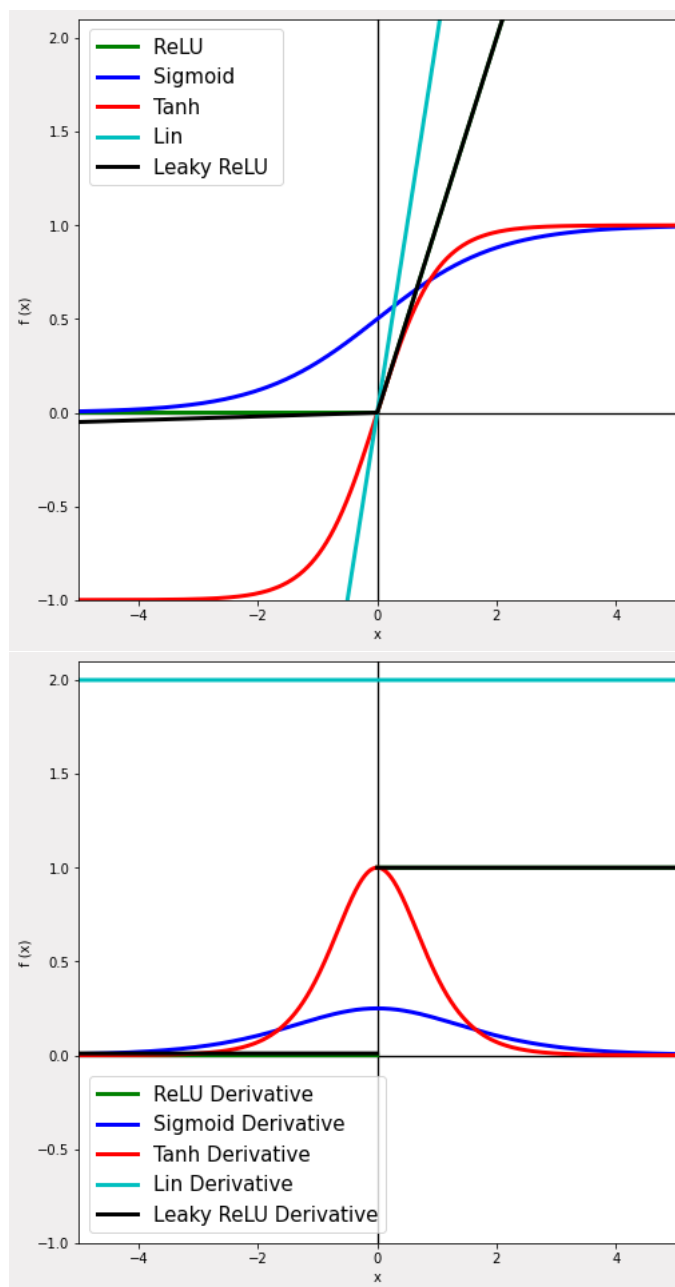
Leaky ReLU je modifikovaná ReLU, ktorá rieši problém odumierania neurónov. Namiesto definovania funkcie ReLU ako 0 pre záporné hodnoty vstupu, ich definuje ako minimálnu lineárnu zložku.

$$f(x) = \max(0, 0.1 \cdot x, x) \quad (3.6)$$

Softmax sa často označuje ako kombinácia viacerých sigmoidov. Ako som spomenul pri popise sigmoidu, sigmoid vracia hodnoty v intervale $[0, 1]$, ktoré možno považovať za pravdepodobnosť údajového bodu patriaceho do konkrétnej triedy. Funkcia Softmax sa hodí v úlohách s klasifikáciou viacerých tried. Na rozdiel od sigmoidu, vráti pravdepodobnosť údajového bodu patriaceho ku každej triede. Súčet pravdepodobností musí byť rovný 1. Táto funkcia sa často používa ako posledná aktivačná funkcia neurónovej siete.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad , \text{ pre } j = 1, \dots, K \quad (3.7)$$

kde K je počet výstupných tried a z_j je hodnota vstupného vektoru na pozícii j .



Obr. 3.2: Priebeh aktivačných funkcií a ich derivácií

V prípade viac-triedovej klasifikácie, bude mať výstupná vrstva toľko neurónov, koľko tried chceme klasifikovať.

3.3 Stratová funkcia (Loss function)

Stratová funkcia je jednou z najdôležitejších častí NN. Jej cieľom je určiť, ako veľmi sa predpoveď nášho modelu líši od správneho výstupu. Výsledná hodnota tejto funkcie sa označuje ako strata (loss). Čím vyššia hodnota straty, tým väčšia odchýlka

medzi našim a správnym výstupom. Každá predpovedaná pravdepodobnosť triedy sa porovnáva so skutočným požadovaným výstupom triedy. Ako jednu z najpoužívanejších funkcií možno považovať Cross-Entropy Loss. V úlohe viac-triedovej klasifikácie (prípád tejto diplomovej práce) sa vypočíta následovne.

$$L_{CE} = - \sum_{i=1}^N t_i \cdot \log(p_i) \quad , \text{ pre } N \text{ tried klasifikácie} \quad (3.8)$$

, kde p_i je Softmax pravdepodobnosť predikcie i -tej triedy a t_i pravdepodobnosť skutočnej triedy.

3.4 Proces tréovania NN

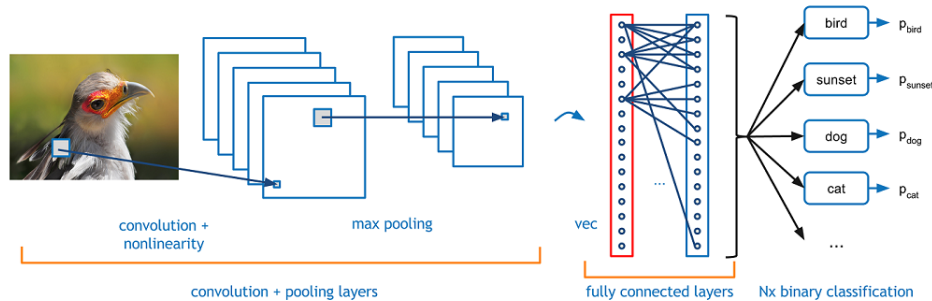
V prípade, že máme na vstupe neurónovej siete obrázok danej veľkosti, každý z pixelov je privedený ako vstup do jednotlivých neurónov vstupnej vrstvy siete. Neuróny sú medzi jednou a následujúcou vrstvou prepojené kanálmi (channels). Každý kanál má priradenú numerickú hodnotu, známu ako váha (weight). Vstupy sú vynásobené odpovedajúcimi váhami a ich suma je poslaná ako vstup neurónov ďalšej vrstvy (skrytá vrstva). Každý z týchto neurónov môže mať priradenú numerickú hodnotu, známu ako bias (odchýlka), ktorý je pripočítaný k sume zo vstupu. Získaná hodnota je následne poslaná do prahovacej funkcie, nazývanej ako aktivačná funkcia (activation function). Výsledok aktivačnej funkcie rozhoduje, či je daný neurón aktivovaný alebo neaktivovaný. Iba aktivované neuróny posielajú dáta do ďalšej vrstvy, opäť pomocou kanálov. Týmto spôsobom sa dáta šíria ako dopredné šírenie (forward propagation). Vo výstupnej vrstve, neurón s najväčšou hodnotou, ktorá je len hodnota pravdepodobnosti, určuje finálny výstup.

V rámci procesu tréovania, okrem vstupných dát neurónovej siete, je poslaný aj skutočný výstup, ktorý sa porovnáva s predpovedaným výstupom. Výsledkom porovnania výstupov je určitá hodnota, ktorá hovorí ako zmeniť výslednú hodnotu pre zmenšenie chyby predikcie. Táto informácia sa prenáša z výstupu na vstup (back propagation). Výsledkom je úprava hodnôt jednotlivých váh.

3.5 Konvolúčna neurónová sieť

Konvolúčna neurónová sieť (Convolution Neural Network, CNN) je jednou z najznámejších variánt neurónových sietí používaných prevažne v počítačovom videní. CNN používané v úlohách počítačového videnia nemajú vstup vo forme vektora hodnôt ale najčastejšie vo forme troj-kanalového obrázku (šírka, výška, hĺbka). Skryté vrstvy CNN pozostávajú z konvolúčných, združovacích (pooling) a plne-prepojených (fully-connected) vrstiev [34].

Neuróny v CNN nie sú spojené so všetkými neurónmi v predchádzajúcej vrstve ale sú spojené len s malou časťou predchádzajúcej vrstvy a majú rovnakú váhu. Základnou jednotkou CNN je filter, označovaný aj ako jadro (kernel). Jadrá sa používajú na extrakciu významných častí zo vstupu pomocou operácie konvolúcie.



Obr. 3.3: Architektúra CNN

3.5.1 Konvolúčna vrstva

Konvolúcia funguje na dvoch signáloch (1D) alebo dvoch obrázkoch (2D). Jeden si možno predstaviť ako vstupný obrázok a druhý ako filter na vstupný obrázok, ktorý vytvára výstupný obraz. Zjednodušene povedané - prijme vstupný signál, vynásobí ho filtrom a získa upravený signál. Výstupom sú tzv. feature maps, ktoré sumarizujú významné vlastnosti vstupu.

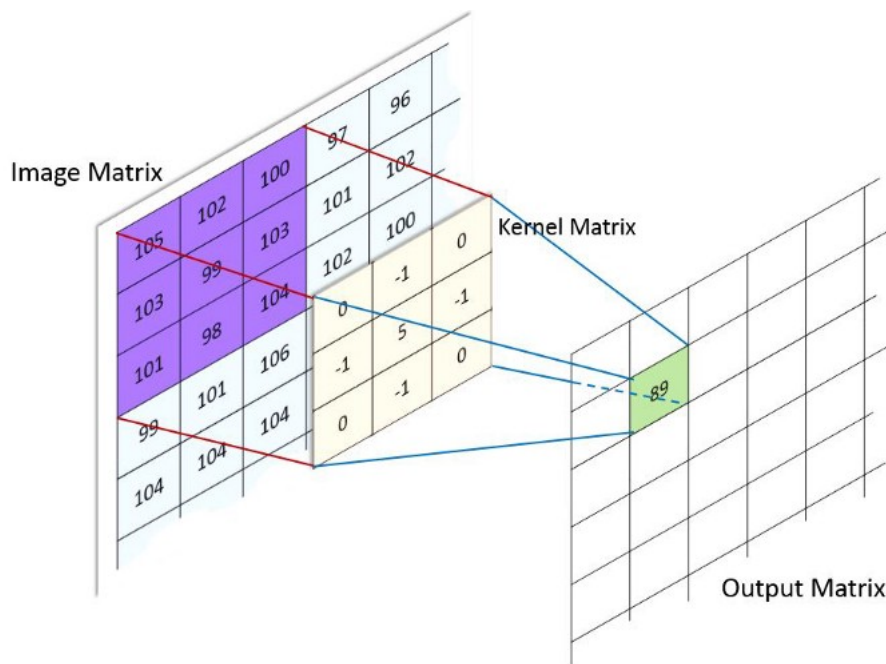
Často používané rozmery filtra sú 3x3, 5x5 alebo 7x7. Ďalšími dôležitými parametrami sú krok (stride) a doplnenie nulami (zero-padding). **Stride** udáva, o akú hodnotu pixelov sa posúva filter po vstupnom obrázku. V prvých vrstvách siete chceme uchovať čo najviac informácií o pôvodnom obrázku. Za týmto účelom môžeme použiť **zero-padding** - doplnenie núl na okraje vstupného obrazu. Rozmery nového obrázku po aplikovaní konvolúcie možno vypočítať:

$$W_2 = \frac{W_1 - F + 2 \cdot P}{S} + 1 \quad (3.9)$$

$$H_2 = \frac{H_1 - F + 2 \cdot P}{S} + 1 \quad (3.10)$$

$$D_2 = K \quad (3.11)$$

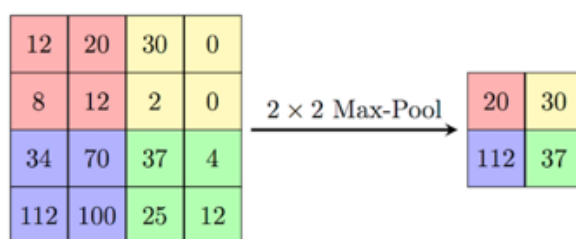
, kde W_1 a H_1 sú rozmery vstupu, F je rozmer jadra, P je rozmer doplnenia núl, S je posun jadra, K je počet filtrov a W_2 , H_2 a D_2 sú rozmery výstupu.



Obr. 3.4: Ukážka 2D konvolúcie [34]

3.5.2 Zdužovacia (pooling) vrstva

Hlavným cieľom pooling vrstvy je podvzorkovať vstup, či už ide o obrázok alebo výstup zo skrytej vrstvy a tým znížiť jeho rozmery a predpokladať vlastnosti obsiahnuté v sub-regiónoch [34]. Zároveň urýchľuje spracovanie dát, lebo znižuje počet parametrov potrebných pre spracovanie. Ako najznámejší typ je označený max-pooling, ktorý z danej oblasti vyberie maximálnu hodnotu. Ďalším typom je average-pooling pre získanie priemernej hodnoty regiónu.



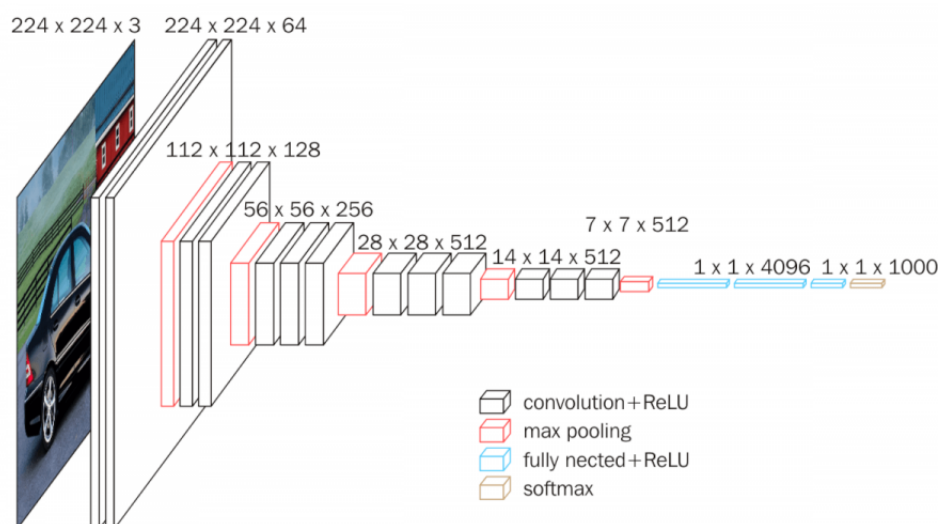
Obr. 3.5: Ukážka funkcie max-pooling [34]

3.5.3 Plne-prepojená vrstva (Fully-Connected)

Ako posledná časť CNN je plne-prepojená vrstva. Na vstupe sa nachádza pretransformovaná feature map-a do stĺpcového vektora. FC vrstvu si možno predstaviť ako ANN z obrázku 3.1.

3.5.4 VGGNet

Visual Geometry Group (VGG) patrí medzi najznámejšie konvolúčne neurónové siete používané v počítačovom videní. Skladá sa z blokov, ktoré pozostávajú z vrstiev 2D konvolúcie a Max Pooling-u. Má dve varianty VGG16 a VGG19, kde čísla udávajú počet vrstiev. VGG16 obsahuje vyše 138 miliónov parametrov [36]. Všetky konvolúčne jadrá sú rozmeru 3×3 a maxpool jadrá 2×2 s krokom 2. Dôvodom pevných rozmerov jadra je, že väčšie filtre môžu byť nahradené viacerými filtermi 3×3 (napr. 5×5 sa môže nahradiť dvoma 3×3 jadrami s dosiahnutím rovnakého výsledku). Tento spôsob vedie k výraznému zníženiu parametrov, čím sa znižuje rýchlosť tréningu a zvyšuje robustnosť. Dôvod, prečo sa nepoužíva VGG s viacerými vrstvami napr. VGG50 je ten, že pri aktualizovaní váh spätným šírením k počiatočným vrstvám, sa gradient násobí lokálnymi gradientami a neustále sa znižuje (vanishing gradient problem), čo výrazne zvyšuje čas tréningu siete. Tento problém je možné vyriešiť tak, že lokálny gradient bude mať hodnotu 1 - použitie ResNet [36].



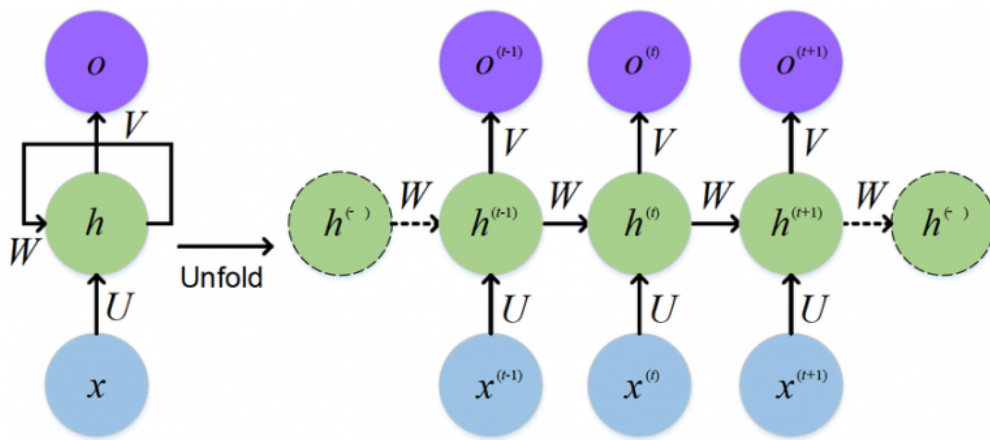
Obr. 3.6: Ukážka architektúry VGG16 [36]

3.6 Rekurentné neurónové siete RNN

Vo všeobecnej NN je vstup spracovaný cez niekoľko vrstiev na výstup s predpokladom, že dva po sebe nasledujúce vstupy sú navzájom nezávislé [34]. Tento predpoklad však nie je pravdivý pre množstvo úloh (predpoveď ceny akcie, predpoveď nasledujúceho slova vo vete). V týchto príkladoch je nutné brať do úvahy závislosť z predchádzajúcich pozorovaní.

RNN vykonávajú rovnakú úlohu pre každý prvok sekvencie, kde výstup závisí od predchádzajúcich výsledkov. Inak povedané, možno si predstaviť, že RNN má akúsi pamäť, ktorá zbiera informácie o tom, čo sa doteraz vypočítalo. V praxi sú obmedzené len na pár spätných krokov. RNN dosahujú výborné výsledky v spracovaní prirodzeného jazyka.

Obrázok 3.7 zobrazuje, ako si možno predstaviť "roztiahnutú" RNN. x^{t-1} , x^t , x^{t+1} a o^{t-1} , o^t a o^{t+1} sú vstupy resp. výstupy v jednotlivých časových okamžikoch. h^{t-1} , h^t a h^{t+1} sú skryté stavy (hidden states) alebo pamäť NN. U , V , W sú váhové matice, ktoré sa učia na tréningových dátach. RNN zdieľa parametre (W , U , V) v rôznych časových krokoch (Parameter Sharing), čo znižuje počet parametrov a tým urýchľuje proces tréningovania [35].



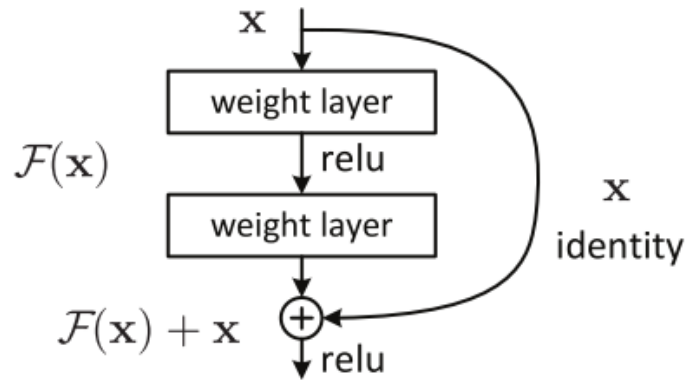
Obr. 3.7: RNN [35]

3.7 Reziduálne neurónové siete ResNet

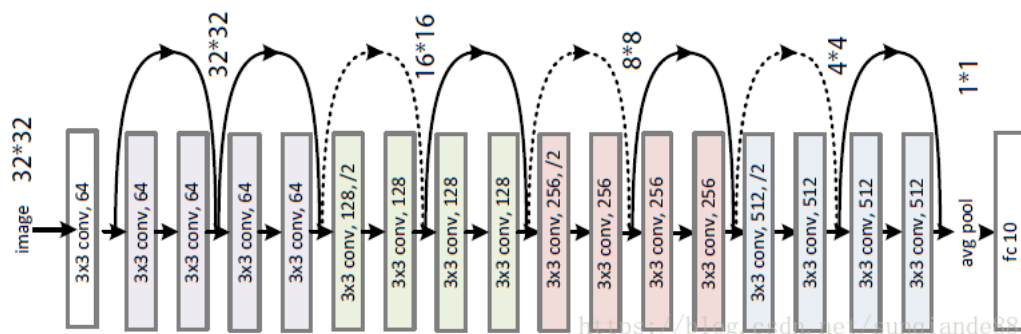
Reziduálna neurónová sieť umožňuje trénovať stovky alebo až tisíce vrstiev a stále dosahuje presvedčivé výkony a výsledky. Základnou myšlienkou architektúry ResNet je zavedenie skratkového spojenia (identity shortcut connection), ktoré preskočí jednu alebo viac vrstiev [37]. Toto spojenie sa označuje ako reziduálny blok. Reziduálny blok rieši problém miznúceho gradientu v príliš hlbokých sieťach. Rovnako ako VGGNet, tiež obsahuje konvolúčne jadrá s rozmermi 3×3 .

Namiesto učenia sa mapovania $x \rightarrow F(x)$ sa sieť učí mapovanie $x \rightarrow F(x) + G(x)$. Keď je rozmer vstupu x a výstupu $F(x)$ rovnaký, funkcia $G(x) = x$ je funkcia identity a skratkové spojenie sa nazýva prepojenie identity (Identity Connection). V prípade, že sa dimenzie $F(x)$ a x líšia (krok > 1 v konvolúčnej medzivrstve) je implementované projekčné prepojenie (Projection connection). Funkcia $G(x)$ zmení rozmery vstupu x na rozmery výstupu $F(x)$. Tieto spojenia sú na obrázku 2.22 vyznačené čiarkovane.

V porovnaní s VGG16, ktorá má 138 miliónov parametrov, má ResNet18 necelých 12 miliónov. Medzi najznámejšie a najčastejšie používané varianty patrí ResNet18, ResNet50 a ResNet101.



Obr. 3.8: Reziduálny blok [37]



Obr. 3.9: ResNet18 [37]

4 Existujúce riešenia

Široké spektrum algoritmov počítačového videnia, ktoré sú používané pre detekciu počasia, analýzu poveternostných podmienok alebo určovanie viditeľnosti možno kategorizovať do štyroch základných skupín - **matematické modely**, **modely založené na filtrovaní**, **modely strojového učenia** a **modely hĺbkového učenia** [41].

1. **Matematické modely** pre analýzu počasia sú primárne používané v aplikáciach detekujúcich hmlu a monitorujúcich stav viditeľnosti s aplikáciami v navigačných systémoch. Autori článku [42] využili MOG (mixture of Gaussians) pre detekciu sneženia a hmly, ktorý je založený na dynamike priestorových a časových rozmerov snímkov. Ďalším možným riešením je využitie Cannyho detektoru hrán. Tento spôsob umožňuje okrem detekovania hmly aj odhadnúť viditeľnosť [43].
2. **Modely založené na filtrovaní** slúžia pre rozpoznávanie poveternostných podmienok zo snímkov na základe ich vizuálnych vlastností a charakteristík. Autori [44] vytvorili model pre detekciu hustej hmly s použitím Gaborovho filtru. Namiesto analýzy lokálnych prvkov, ako napríklad, označenie jazdných pruhov, alebo cestnej oblasti, použili globálne prvky z hladiska spektra Fourierovej transformácie [44].
3. **Modely strojového učenia** priniesli pokrok v rozpoznávaní viacerých stavov alebo tried počasia súčasne. Autori [46] v ich práci zameranej na klasifikáciu poveternostných podmienok využili model SVM (Support Vector Machine), ktorý natrénovali na jednofarebných obrázkoch. Tento model je však limitovaný iba na tri stavy počasia, a to hustý dážď, jemný dážď a jasné počasie. Ďalším príkladom použitia strojového učenia je práca [47]. Autori sa zamerali na extrakciu viacerých prvkov ako je napríklad obloha, snehové vločky, dažďové kvapky, tieň, spoliehajúc sa na k-najbližších susedov (k-nearest neighbours) a SVM.
4. **Modely hĺbkového učenia** sú založené hlavne na modeloch konvolúčných neurónových sieti (CNN). Tento prístup ukázal pokroky v úlohách spracovania obrazu a uvedomovania si scény. Rovnako boli postupne vyvinuté rôzne modely, ktoré na základe konvolúčnej štruktúry hĺbkového modelu dokážu klasifikovať počasie do konkrétnych stavov. Práca [48] používa CNN model, ktorý dokázal vyriešiť problém kombinácie viacerých tried počasia v jednom obrázku - model dokáže odvodiť viac ako jednu triedu v závislosti od charakteristiky vstupného obrazu. Ďalším príkladom je paralelný hĺbkový CNN model WeatherNet [41], ktorý je zložený zo štyroch hĺbkových CNN modelov. Detekuje úsvit/súmrak, deň/noc, zrážky(dážď, sneh alebo bez zrážok) a hmlu.

Tab. 4.1: Zhrnutie výhod a nevýhod modelov

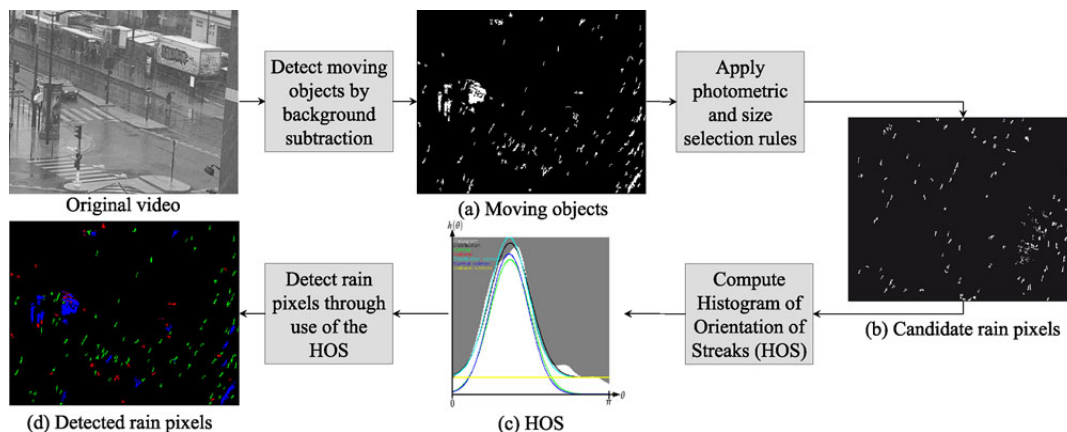
	Matematické modely	Modely založené na filtrovaní	Modely ML	Modely DL
Výhody modelu	jednoduchosť, minimum dát	jednoduchosť, minimum dát	presnosť, rýchlosť, nižšia komplexnosť ako DL modely	presnosť, minimálne preddefinované nastavenia, prenosové učenie
Nevýhody modelu	presnosť, závislé na konkrétnych parametroch, určené pre špecifickú úlohu, nízka opätovná použitelnosť	závislosť na hodnotách pixelov, závislé na konkrétnych parametroch, určené pre špecifickú úlohu, nízka opätovná použitelnosť	určené pre špecifickú úlohu, veľkosť datasetu	časová náročnosť natrénovania modelu, potrebné veľké množstvo dát, zložitá interpretácia modelu, komplexnosť

4.1 Detekcia zrážok vo vzduchu

4.1.1 Prvá metóda detekcie zrážok

Prvým krokom tohto riešenia [38] je segmentácia potenciálnych dažďových pásov pomocou klasickej metódy odčítania pozadia. Najbližšie kvapôčky nie sú zaostrené, zatiaľ čo tie najvzdialenejšie zaostrené sú. Kvapky, ktoré sú zaostrené, sú rozmazané pohybom a vytvárajú v obrázkoch pruhy. Pretože zaostrené kvapôčky sú najvzdialenejšie, sú pruhy väčšinou malé. Pri segmentácii týchto pruhov autori predpokladali, že kvapôčka je malý, podlhovastý, pohybujúci sa objekt, ktorý je jasnejší ako jeho pozadie.

Na segmentáciu pohybujúcich sa objektov boli navrhnuté rôzne metódy. V kontexte vizuálneho sledovania sa bežne používajú prístupy založené na modelovaní pozadia [50], pretože umožňujú zohľadniť postupné zmeny osvetlenia v scéne neustálou aktualizáciou modelu pozadia. Takéto metódy v zásade umožňujú výpočet



Obr. 4.1: Postup algoritmu [38]

modelu pozadia (BG-background) scény, ktorý obsahuje statické objekty, ako aj modelu popredia (FG-foreground) scény, ktorý obsahuje pohybujúce sa objekty. Autori zvolili populárny prístup Mixture of Gaussians (MoG).

Posledný krok segmentácie dažďových pruhov spočíva v odfiltrovaní príliš veľkých alebo príliš malých objektov v popredí. Výsledkom je, že sa odstránia pohybujúce autá, stromy a podobne.

Výsledkom kroku segmentácie je obrázok s malými oblasťami odpovedajúcimi dažďovým pruhom alebo šumom. Predpokladá sa, že väčšina odpovedá dažďovým pruhom s takmer vertikálnou orientáciou. Pre výpočet orientácie bola použitá metóda používajúca geometrické momenty (HOS).

Ak je na scéne dážď, povrch Gaussovho rozdelenia by mal byť dostatočne vysoký. Ak na scéne nie je dážď, povrch Gaussovho rozdelenia by mal zostať väčšinu času dosť nízky. Hodnoty parametrov rozhodovacieho procesu boli vybrané pomocou procesu pokus omyl.

Potom zostavíme takzvaný histogram orientácie pruhov (HOS) akumuláciou orientácií rôzne spojených komponentov získaných metódou geometrických momentov. Získané data z histogramu sú modelované Gaussian-uniform mixture.

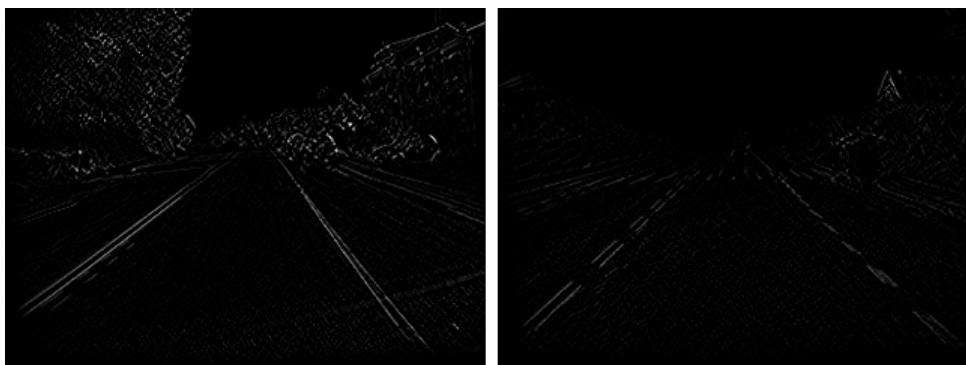
4.1.2 Druhá metóda detekcie zrážok

Podľa optických fyzikálnych charakteristík dažďových kvapiek a snehových vločiek autori [39] študovali algoritmus detekcie kvapiek a vločiek, založený na klasickom rozdieli frame-ov videa. Na základe algoritmu detekovali počiatočné, snehové binárne spojené oblasti. Na rozlíšenie snehových a dažďových regiónov pridali obmedzenia oblasti a smerového uhlu. Nakoniec použili tri algoritmy rozdiel dvoch frame-ov, rozdiel troch frame-ov a rozdiel piatich frame-ov na odstránenie dažďa resp. sneženia z obrázkov.

4.2 Detekcia hmly

4.2.1 Prvá metóda detekcie hmly

Podľa [40] je dôležitým efektom hmly v obrázkoch redukcia vysoko frekvenčných zložiek, ktoré je možné merať metódami detekcie hrán. Obrázky, v ktorých sa vyskytuje hmla, sú rozmazanejšie a majú nižší kontrast ako obrázky slnečného počasia. To znamená, že hlavne v hornej časti obrázku s hmlou je nižšia informácia vo vyšších frekvenciách. Čím je hrana ostrejšia, tým vyššia je úroveň šedej na výstupnom obrázku. Táto práca použila dataset nafotený kamerou na čelnom skle automobilu.



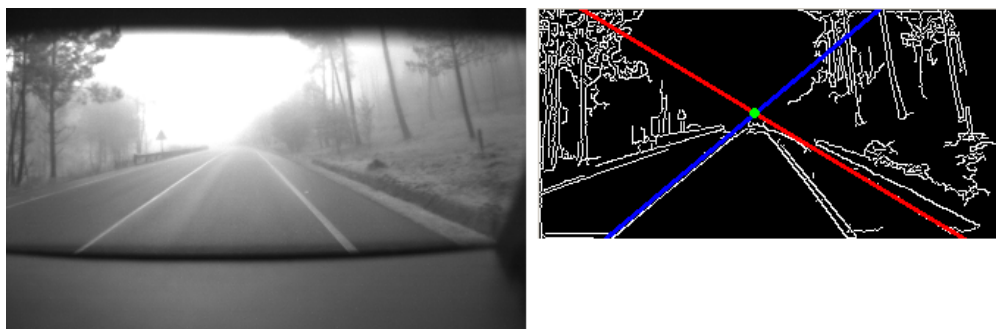
Obr. 4.2: Efekt hmly na hrany [40]

V prípade, že sa jedná o obrázok s hmlou, Cannyho detektor je použitý pre zvýraznenie a filtráciu hrán - ponechanie iba najvýraznejších. Potom je použitý Houghov detektor čiar, ktorý má určiť okrajové čiary vozovky. Tieto okrajové čiary vozovky slúžia pre výpočet úbežného bodu (vanishing point). Úbežným bodom obrazu je bod, kde sa v obraze krížia rovnobežné čiary, ktoré nie sú kolmé na optickú os.

Potom sú tieto okrajové obrázky vypočítané pomocou Hough detektora čiar, ktorý sa používa na odhad okrajových čiar cesty. Filtrovaním riadkov nájdených detektorom Hough, aby sa dostatočne oddelili, a získaním jeho parametrov sa nachádza úbežný bod obrazu.

Ďalším krokom algoritmu je segmentácia založená na metóde rastúcich regiónov (growing regions) pre nájdenie hranice medzi vozovkou a oblohou. Pre vylepšenie segmentácie oblohy je použitá dilatácia. V prípade segmentácie cesty je použitá erózia.

Odhad viditeľnosti v prípade prítomnosti hmly je založený na predpoklade, že ak je na obrázku zaznamenaná hmla, časť vysegmentovanej oblohy sa dostane do časti vysegmentovanej vozovky - výška oblohy je menšia. Ak sa hmla nenachádza, výška oblohy a výška cesty odpovedajú výške úbežného bodu. Odhad viditeľnosti je možné určiť pomocou rovníc projekcie kamery.



Obr. 4.3: Nájdenie inflexného bodu [40]

4.2.2 Druhá metóda detekcie hmly

Táto metóda sa zaoberá návrhom pokročilého asistenčného jazdiaceho systému. Metóda autorov [43] je podobná predchádzajúcej popísanej metóde detekcie hmly, je taktiež založená na jednej kamere vo vozidle. Autori sú schopní detekovať dennú hmlu v širokej škále rôznych scén. Okrem detekcie hmly, dokážu odhadnúť aj viditeľnosť. Táto metóda vychádza z Koschmiederovho zákona, ktorý popisuje vzťah medzi zdanlivým kontrastom objektu pri pohľade proti oblohe vzdialeným pozorovateľom a jeho prirodzeným kontrastom - kontrast, ktorý by daný objekt mal oproti horizontu pri pohľade z veľmi krátkej vzdialenosti.

Algoritmus je tvorený nasledujúcimi krokmi. Najprv sa použije Cannyho detektor hrán na vstupný obraz. Ďalej sú vykonané kroky odhadu horizontálnej čiary a odhad inflexného bodu. Odhad horizontálnej čiary bol prevedený na základe vlastností obrazu - je ale potrebné prejsť vozidlom aspoň malú vzdialenosť. Pre výpočet inflexného bodu sa musí nájsť - maximálne pásmo, ktoré pretína oblasť v obrázku, ktorá predstavuje minimálnu zmenu gradientu. Ak nie je nájdené takéto pásmo, je možné vyhlásenie, že hmla na obrázku nie je prítomná.

4.2.3 Tretia metóda detekcie hmly

Tento článok [44] sa venuje detekcii hmly z kamery namontovanej na automobile. Jeho hlavná myšlienka spočíva v použití deskriptorov obrazu a následnej klasifikácie na odlíšenie prítomnosti a neprítomnosti hmly. Tieto deskriptory obrázkov sú globálne a popisujú celý obrázok použitím Gaborovho filtru pri rôznych frekvenciách, mierkach a orientáciách.

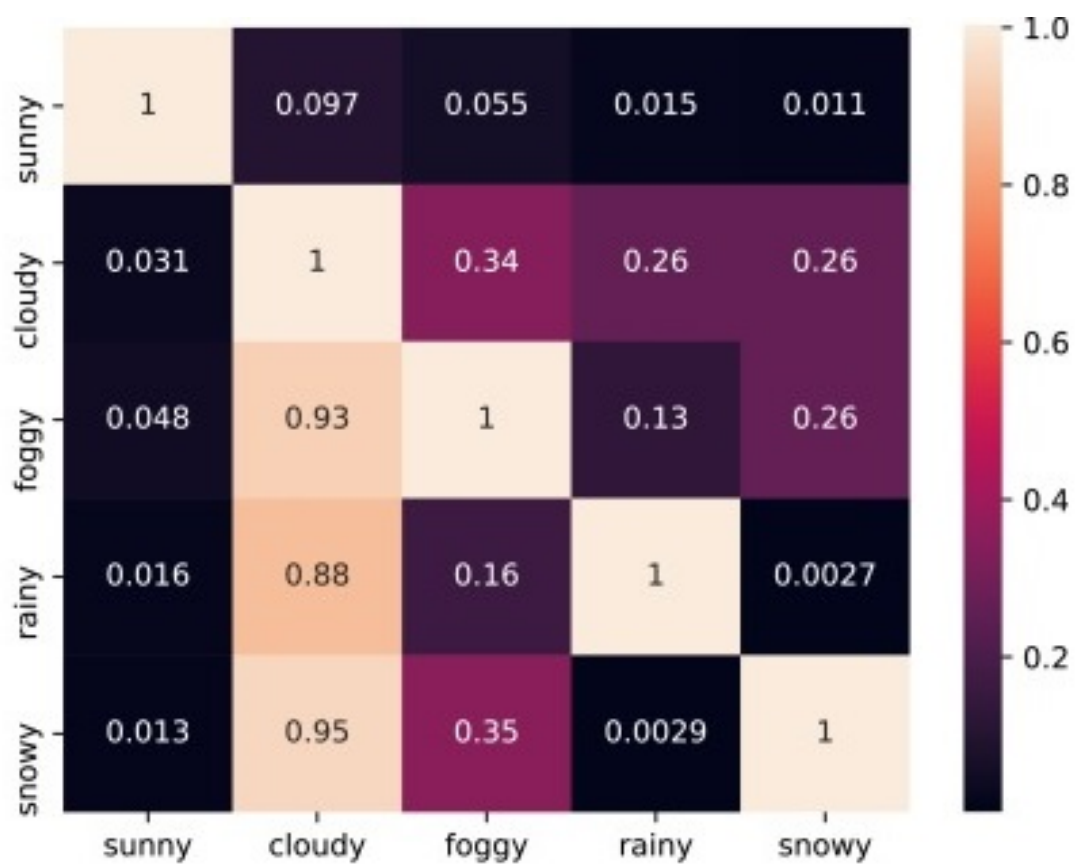
V prípade scény s hmlou sa frekvenčné zložky sústreďujú pri nízkych frekvenciách, zatiaľ čo na scéne bez hmly sa nachádza širšie spektrum. Dôvodom je zoslabenie kontrastu a rozmazanie hrán vplyvom hmly. Ostré hrany sú tvorené veľkým množstvom nízkych a vysokých frekvencií, rozmazané hrany pozostávajú iba z nízkych.

Aby sa vyhli tomu, že niektoré obrazové oblasti dominujú v spektre, a aby sa znížili efekty osvetlenia, najskôr je vykonaná normalizácia obrazu. Ďalším krokom je extrakcia vlastností aplikovaním Fourierovej transformácie a redukciou vlastností použitím Gaborovho filtru. Posledným krokom je klasifikácia pomocou podporných vektorov (SVM).

4.3 Viac-triedové rozpoznávanie počasia

Autori [48] sa pokúsili navrhnuť model, ktorý dokáže klasifikovať aj viac ako jednu prítomnú triedu na obrázku (multi-label classification) použitím CNN-RNN architektúry. CNN bola použitá na extrakciu čo najviac predbežných znakov z obrázku. RNN ďalej spracováva vlastnosti a zisťuje závislosti medzi jednotlivými triedami počasia.

Použité obrázky boli pred trénovaním zmenšené na veľkosť 256 x 256 pixelov. Následne bola prevedená augmentácia dát (náhodne pootočené, náhodné orezanie, pridanie náhodného šumu). Použitý dataset obsahoval 10000 obrázkov z piatich tried (slnečno, oblačno, hmla, dážď a sneženie). Dataset obsahuje rôzne scény - dedina, mesto, a podobne.



Obr. 4.4: Dosiahnuté výsledky CNN-RNN [48]

5 Vstupné dáta

Táto kapitola práce sa venuje popisu zisku a príprave vstupných obrazových dát, ktoré sú ďalej spracovávané a analyzované algoritmami spracovania obrazu. Obrazové dáta si možno predstaviť ako obrázky alebo sekvencie obrázkov nasnímané kamerou.

Na základe skutočnosti, že je moja diplomová práca zameraná na detekciu zrážok vo vzduchu, detekciu hmly a detekciu zrážok na vozovke, som vyskladal tri datasety - dataset zachytávajúci počasie obecné, dataset zachytávajúci stav vozovky a dataset zachytávajúci počasie v doprave. V rámci ďalších kapitol práce, ktoré sa venujú algoritmom a návrhu riešenia, som sa snažil vybrať čo najpestrejšie ukážkové obrázky z dopravy, ktoré znázorňujú rôzne situácie a polohy kamier.

5.1 Dataset stav vozovky

Tento dataset je prispôbený na dopravnú scénu tak, aby primárne zaznamenával stav vozovky (suchá, mokrá, zasnežená) bez ohľadu na okolitú scénu. Vozovku som si nafotil sám a obsahuje niekoľko desiatok fotiek, ktoré som použil pri segmentácii vozovky a následnej detekcii zrážok na vozovke resp. určení stavu vozovky. Možno ho označiť ako dočasný, pomocný dataset, ktorý som mimo ladenia metód týkajúcich sa vozovky nepoužil.



Obr. 5.1: Ukážka datasetu stavu vozovky

5.2 Dataset počasia s obecnou scénou

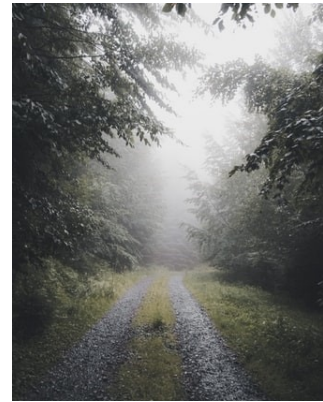
Jedná sa o dataset, ktorý zachytáva ľubovoľný stav počasia v ľubovoľnej scéne (príroda, doprava, ľudské obydlie a iné). S rýchlym pokrokom počítačového videnia sú

aj pre verejnosť čoraz dostupnejšie rozsiahle datasety, a preto prevažnú časť obecného datasetu som prevzal z dostupných zdrojov na internete [16] a zvyšok som si, podľa možnosti, nafotil sám. Finálna podoba datasetu je následovná - dataset je rozdelený do piatich tried, podľa stavu počasia (slnečno, oblačno, hmliсто, daždivo a sneženie). Každá trieda obsahuje približne 4000 obrázkov, čo spolu tvorí 20000 obrázkový dataset.

Dataset počasia s obecnou scénou som vďaka jeho pestrosti a množstvu použil pri testovaní a ladení algoritmov detekcie dažďa, snehových vločiek a hmly, a primárne na tréňovanie modelu neurónovej siete a počiatočné testovanie nápadov rozpoznávania počasia, ktorým je venovaná ďalšia kapitola. Ako som spomenul, výhoda obecného datasetu spočíva v ľahkej dostupnosti, obrovskému rozsahu a tým aj pestrosti a komplexnosti, čo je konkrétne v tejto práci, ktorá je hlavne zameraná na dopravnú scénu súčasne aj nevýhodou.



(a)



(b)



(c)



(d)

Obr. 5.2: Ukážka datasetu počasia s obecnou scénou [16]

5.3 Dataset počasia s dopravnou scénou

Posledný dataset je kombináciou predchádzajúcich datasetov. Obsahuje snímky s dopravnou scénou (snímanie vozovky) a zároveň zobrazuje stav počasia. Z toho vyplýva, že je pre použitie v tejto práci najideálnejší dataset. Veľkú časť dát mi dodala moja skoliťka. Tieto snímky nemôžem zverejniť v textovej práci lebo patria do vlastníctva tretej strany, od ktorej nemám povolenie k zverejneniu. Napriek tomu som ich použil na testovanie a vyhodnotenie úspešnosti navrhnutých metód a sú súčasťou priloženého CD. Snímky sú z troch reálnych dopravných kamier z oblasti ČR. Druhú časť datasetu som prebral z internetu [18] a zároveň opäť v rámci možností aj niečo sám nafotil. Oproti obecnému datasetu, ktorý zachytáva päť stavov počasia, tento prípad obsahuje prevažne dáta z tried daždivo, sneženie a hmliсто. Dôvodom je, že v prípade snímania vozovky zvykne byť kamera pod určitým uhlom, čo môže viesť k dôsledku absencie oblohy (prípade troch dodaných stanovísk). Vo finálnej verzii má dataset 807 obrázkov. Túto sadu obrázkov som použil na finálne otestovanie a vyhodnotenie úspešnosti navrhnutých metód.



(a)



(b)



(c)



(d)

Obr. 5.3: Ukážka datasetu počasia s dopravnou scénou [18]

6 Návrh a realizácia použitých algoritmov

Zo získaných informácií a prehľadu o existujúcich riešeniach analýzy počasia z obrazových dát, sa v nasledujúcich kapitolách budem venovať návrhu algoritmov, ktoré použijem na monitorovanie počasia. V úvode kapitoly popíšem, ako som postupoval pri návrhu vhodných algoritmov. Ako prvý krok som rozdelil úlohu na tri podúlohy alebo moduly, ktoré nezávisle riešia danú problematiku, a to segmentácia vozovky a zároveň určenie stavu vozovky, detekcia prítomnosti hmly a detekcia zrážok vrátane parametrizácie. Pri návrhu konkrétnych algoritmov som sa snažil vychádzať hlavne z toho, ako človek vníma jednotlivé stavy počasia a aké zmeny nastávajú pri prechodoch z jedného stavu do druhého napr. zmena rozloženia farieb v scéne, zmena kontrastu, zníženie ostrosti hrán objektov v pozadí a podobne. Aj keď jednotlivé moduly riešia samostatne rôzne stavy počasia, ich výsledky sa dajú kombinovať a tým dosiahnuť presnejšie výsledky. Na základe kombinácie uvedených príznakov je možno rozlišovať základné druhy počasia ako slnečno, zamračene, hmlisto, daždivo a sneženie. Tieto metódy možno považovať za klasické metódy používané v počítačovom videní. Ďalšia časť kapitoly je venovaná prístupu riešenia problémov počítačového videnia použitím neurónovej siete. Použil som dve rôzne architektúry, konkrétne konvolučnú neurónovú sieť VGG16 a reziduálnu neurónovú sieť ResNet50.

6.1 Rozbor možných prístupov

6.1.1 Segmentácia vozovky a určenie stavu vozovky

Aby som mohol určiť aktuálny stav vozovky, potrebujem vedieť, kde sa na obrázku nachádza a oddeliť ju od zvyšku obrazu - vysegmentovať. Vozovka môže byť na obrázku zachytená ako rovný, priamy úsek bez zatačky alebo zakrivený úsek s jednou alebo viacerými zatačkami. Ako ďalší potenciálny bod, ktorý môže pomôcť pri segmentácii je krajnica. Pri udržiavaných cestách môže dopomôcť odlíšiť vozovku od oblasti mimo vozovky nakreslená biela čiara alebo hranica asfaltu a oblasti už mimo vozovky. Vozovka väčšinou zaberá najväčšiu časť obrázka v jeho spodnej polovici. Na základe popísaných myšlienok o možných tvaroch vozovky a jej umiestneniu som implementoval tri rôzne prístupy segmentácie vozovky. Po úspešnom oddelení vozovky od zvyšku obrázka som sa mohol zamerať na určenie stavu vozovky. Hlavná myšlienka analýzy vozovky spočíva, ako sa zmení farba asfaltu za rôzneho počasia. Suchá vozovka má iné rozloženie farieb ako mokrá, a podobne mokrá vozovka má iné farby ako zasnežená. Implementoval som jednu metódu analýzy vozovky. Segmentácia vozovky umožňuje analyzovať len tú oblasť, ktorá sa týka samotnej vozovky.

6.1.2 Detekcia zrážok

V práci som uvažoval dva typy zrážok - dážď a sneženie. Zrážky zachytené na obrázkoch si možno v ideálnom prípade predstaviť ako malé objekty, ktoré majú v danom čase veľmi podobnú veľkosť, tvar, farbu a smer pohybu (ak zanedbáme vplyv vonkajších javov ako napr. vietor). V skutočnosti zrážky zachytené v obraze vytvárajú tenké pásy. Vzhľadom k tomu, že zrážky sú blízko objektívu kamery, sa budú na scéne pohybovať rýchlejšie ako iné objekty. Pre detekciu zrážok som implementoval dve metódy.

Detekcia dažďa je náročná úloha, pretože voda je takmer priehľadná lebo molekuly vody nedokážu absorbovať veľké množstvo viditeľného svetla. Inak povedané, objekty za dažďovými kvapkami sú viditeľné. Z tohto dôvodu je potrebná vhodná voľba optiky kamery a zároveň pozorovacej vzdialenosti, aby bolo možné zachytiť dostatok pohybu spojeného s padajúcimi kvapkami vody. Dážď sa dá na obrázku detekovať aj nepriamo napr. ztmavnutím niektorých objektov v scéne (napr. mokrá vozovka alebo strechy budov).

Detekcia sneženia je o niečo jednoduchšia úloha ako detekcia dažďa. Snehové vločky majú zvyčajne väčšiu veľkosť ako kvapky dažďa. Podstatnejším rozdielom je, že snehové vločky nie sú priehľadné - objekty podobnej veľkosti ako snehová vločka nie sú viditeľné za snehovými vločkami. Rovnako ako dážď, aj sneženie je možné detekovať aj nepriamo - zasnežené (biele) okolie.

6.1.3 Detekcia prítomnosti hmly

Ak sa v obraze nachádza hmla, prejaví sa to všeobecným rozmazaním obrazu, ktoré možno prirovnáť k nízkopriepustnej filtrácií. Ďalším znakom prítomnosti hmly je zníženie celkového kontrastu v obraze z dôvodu, že veľká časť obrazu je "zakrytá" sivým oblakom hmly. Oblak je sivej farby a tým, že rozmazáva obraz, kontrast klesá. Vplyvom hmly sa taktiež zmenší celkový počet hrán v obrázku a zároveň tie čo ostanú, budú slabšie. Posledným prejavom hmly, ktorý som uvažoval, je zmena frekvenčného obsahu obrazu. Čím je hmla hustejšia, tým je kontrast menší, celkový počet hrán v obraze je menší a hrany, ktoré v spodnej časti obrazu ostali, sú zoslabnuté resp. rozmazanejšie. Hmla sa na obrázku zvyčajne vyskytuje v hornej polovici. Vplyv hmly na obraze možno vidieť na obrázkoch 6.1.

Pre spoľahlivé určenie, či sa na obraze nachádza hmla som implementoval štyri metódy. Každá z nich sa pozerá na úlohu z trošku iného uhla a s kombináciou výsledkov je možné detekovať hmlu, parametrizovať viditeľnosť a rozlíšiť slnečné počasie a zamračené počasie. Spôsob odhadu viditeľnosti je popísaný v ďalšej kapitole.

Implementoval som štyri metódy, ktoré dokážu detekovať prítomnosť hmly, odhadnúť aktuálnu viditeľnosť a odlíšiť hmlu od zamračeného a slnečného počasia.



(a)



(b)



(c)

Obr. 6.1: Ukážka porovnania scény s hmlou a bez hmly [51] a ukážka sneženia

6.2 Realizácia algoritmov

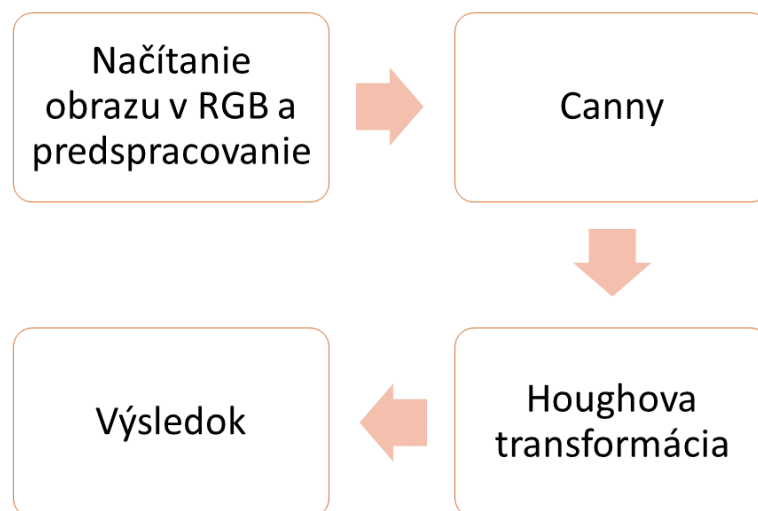
Na základe rozboru a analýzy možných prístupov sa nasledujúca časť kapitoly bude venovať popisu tvorbe algoritmov. Všetky algoritmy klasických metód sú napísané v jazyky C++ s čo maximálnym možným využitím knižnice OpenCV.

6.2.1 Segmentácia vozovky a analýza jej stavu

Z naštudovania existujúcich riešení a prevedeného rozboru som implementoval tri rôzne metódy segmentácie vozovky.

Segmentácia vozovky - metóda Houghova transformácia

Prvá metóda je založená na predpoklade, že vozovka je priama oblasť (bez zatačiek) na obraze. Hlavnou myšlienkou tejto metódy je nájdenie priamok, ktoré oddeľujú vozovku od oblasti mimo vozovky. Ako hlavný záchytný bod som uvažoval biele čiary na krajnici. Avšak stále sú aj také cesty, ktoré biele čiary nemajú a v takýchto prípadoch som musel pracovať so samotnou hranicou vozovky.



Obr. 6.2: Postup algoritmu segmentácie založenej na hľadaní priamok

Aby algoritmus mohol niečo robiť, potrebuje mať na vstupe nejaké dáta. Ako vstup sa načíta troj-kanálový obraz, ktorý sa predspracuje. Krok predspracovania spočíva v zmenšení pôvodného obrázku s dôrazom na zachovanie pomeru výšky a šírky pôvodnej veľkosti obrazu. Ako najčastejšie používané rozmery po zmenšení som použil hranicu väčšej strany 500 pixelov.

Následuje krok hľadania hrán v obraze použitím Cannyho detektoru. Na rýchle pripomenutie z kapitoly o spracovaní obrazu, Canny pracuje v štyroch krokoch: filtrácia šumu, nájdenie hrán, ztenšenie hrán a dvojúrovňové prahovanie. Úlohou tohto kroku algoritmu je nájsť najvýraznejšie hrany na vozovke, na ktorých závisí následujúci krok a previesť obrázok do binárnej podoby.

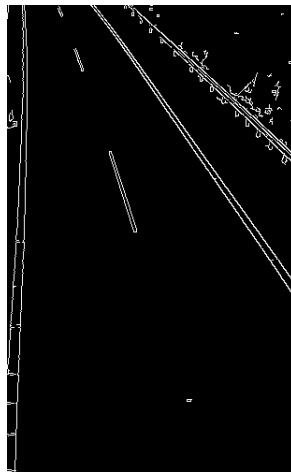
Posledným krokom je Houghova transformácia, ktorá pracuje s binárnym obrazom. Jej cieľom je vyhľadať priamky, ktoré môžu byť potenciálnou hranicou vozovky. Získané priamky sú následne vykreslené do pôvodného obrazu, ktorý sa uloží ako výstup algoritmu.

Obrázky 6.3 a 6.4 znázorňujú grafický výstup metódy založenej na Houghovej transformácii. Obrázok a) znázorňuje vstupný originálny trojkanálový obraz, b) výsledok po použití Cannyho detektoru, binárny obraz a c) výstup ako zakreslenie nájdených priamok do vstupného obrazu.

Oba obrázky zachytávajú dataset stav vozovky. Vozovku možno v oboch prípadoch brať ako rovnú bez zatačiek. Obrázky porovnávajú, ako Houghova transformácia reaguje na okolitú scénu vozovky. V prvom prípade tvorí vozovka veľkú časť obrázku, druhý prípad je komplexnejší, obsahuje dva prúdy diaľnice, medzi nimi be-



(a) Pôvodný obrázok v RGB



(b) Výstup z Canny detektoru ako binárny obrázok

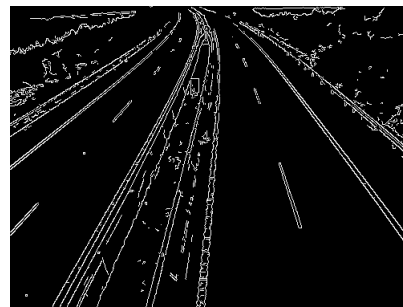


(c) Výstup algoritmu detekcie vozovky Houghovou transformáciou

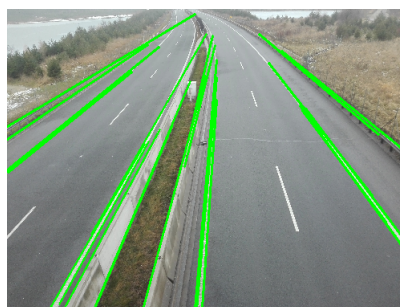
Obr. 6.3: Ukážka metódy segmentácie s použitím Houghovej transformácie



(a) Pôvodný obrázok v RGB



(b) Výstup z Canny detektoru ako binárny obrázok



(c) Výstup algoritmu detekcie vozovky Houghovou transformáciou

Obr. 6.4: Ukážka č.2 metódy segmentácie s použitím Houghovej transformácie

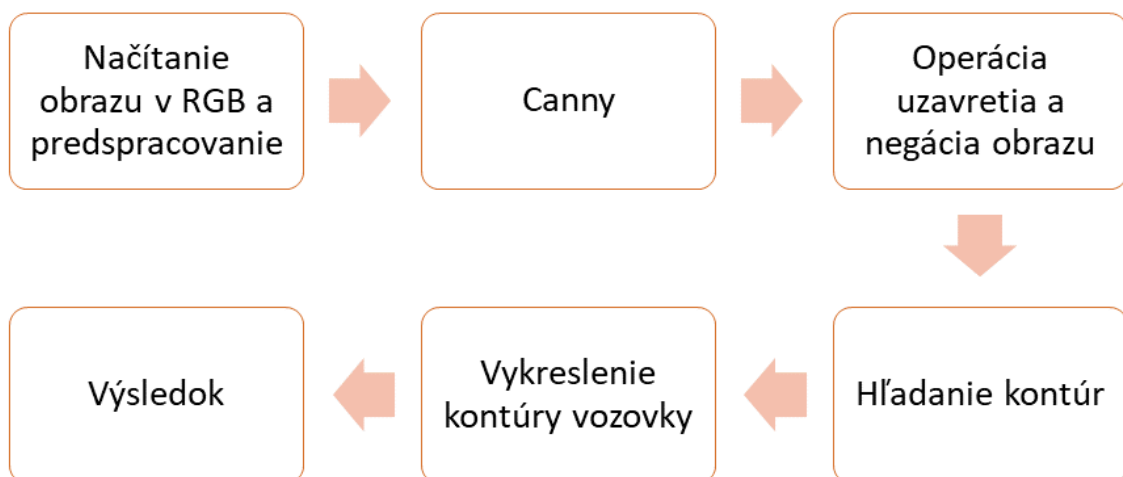
tónové panely a popri ceste je viac prítomná okolitá scéna ako pri predchádzajúcom obrázku.

Z uvedených grafických výstupov vidno, že táto metóda je na detekciu vozovky ako-tak použiteľná len vo veľmi špecifických prípadoch - pohľad kamery priamo na vozovku s čo najväčšou elimináciou častí, ktoré s vozovkou nesúvisia, čo výrazne limituje použitie metódy. Pre úlohu segmentácie vozovky sa Houghova transformácia javí ako nevhodné riešenie. Na obrázku 6.4 možno vidieť, že okrem vozovky sú detekované rôzne časti pozadia a oblasti mimo vozovky. Pre vylepšenie detekcie na podobných obrazoch ako 6.3 je vhodné použiť hypotézu, že sa vozovka s určitou istotou nachádza na spodku obrazu a má tvar lichobežníka, poprípade trojuholníka. Tvar, veľkosť a smer masky závisí od konkrétnej scény a vymyslieť všeobecnú masku pre ľubovoľnú scénu sa javí ako veľmi náročná úloha. Po prejdení takouto maskou po celom obrázku by bolo možné výrazne odfiltrovať nežiadúce detekované priamky. Metódu s Houghovou transformáciou na základe vyššie uvedených skutočností nepovažujem za vhodnú metódu segmentácie vozovky.

Segmentácia vozovky - metóda Cannyho detektoru

Po nedostatočných výsledkoch segmentácie metódou Houghovej transformácie som sa zamyslel, čo by sa z nej dalo použiť a čo je potrebné zmeniť pre dosiahnutie lepších výsledkov. Výsledkoch z Cannyho detektoru hrán však vykazovali zaujímavý potenciál. Aj v tomto prípade riešenia som stále uvažoval oddelenie vozovky od jej okolia pomocou bielych čiar alebo samotnej hranice vozovky. Nespoliehal som sa však na to, že vozovka je na obraze zobrazená ako priamy úsek.

Prvý krok algoritmu je rovnaký ako pri predchádzajúcej metóde - zmenšenie pôvodného obrazu na veľkosť maximálne 500 pixelov. Druhý krok taktiež ostal rovnaký, a teda prevod trojkanálového obrazu na binárny a zároveň nájdenie najvýznamnejších hrán v obraze použitím Cannyho detektoru hrán.



Obr. 6.5: Postup algoritmu segmentácie založenej na Cannyho detektore

Výsledný obraz z Cannyho detektoru obsahuje hrany so šírkou iba jedného pixelu. V prípade, že obraz obsahoval pestrú scénu a okolie popri vozovke, bolo nutné použiť vyššie hodnoty prahov hysterézneho prahovania, čo môže viesť aj k prerušeniu hrany hranice vozovky.

Z toho dôvodu sú ako tretí krok algoritmu použité morfologické operácie, konkrétne operácia uzavretia. Opäť pre rýchle pripomenutie, uzavretie je operácia dilatácie, ktorú nasleduje operácia erózie. Uzavretie rozšíri danú bielu oblasť, vyplní prázdne miesta v bielej oblasti a hlavne, môže spojiť blízke oblasti, ktoré boli rozpojené vplyvom použitia vysokých prahových hodnôt.

Po zvýraznení hrán by mala byť vozovka úspešne oddelená od okolitej scény. Problémom pre ďalšie pokračovanie algoritmu je, že oblasť záujmu, kde sa nachádza vozovka je čierna, a biele sú len jej hranice. Inak povedané, hľadaný objekt musí byť biely a jeho pozadie čierne. Z toho dôvodu, je nutné aktuálny obraz znegovať - čierne oblasti budú biele a naopak, biele oblasti čierne.

Ako posledný krok algoritmu je vyhľadávanie kontúr. Kontúry možno popísať ako spojitú krivku, ktorá spája všetky spojité body pozdĺž hranice, ktorá má rovnakú farbu resp. intenzitu. Pre tento účel bola použitá funkcia z OpenCV `findContours()`. Algoritmus hľadanie kontúr je založený na hierarchickom vzťahu medzi hranicami. Algoritmus prechádza po obraze z ľavého horného rohu po pravý dolný roh, pokiaľ nenarazí na pixel, ktorý je vonkajšou hranicou alebo hranica otvoru. Vonkajšia hranica je definovaná ako $F_{i,j} = 1, F_{i,j-1} = 0$ a hranica otvoru ako $F_{i,j} \geq 1, F_{i,j+1} = 0$. Pixel, ktorý splňuje podmienku je označený ako začiatok vonkajšej hranice a je mu priradené unikátne poradové číslo. Počas prechodu obrazom je ponechané aj poradové číslo poslednej hranice pre určenie nadradenej hranice. Algoritmus pokračuje po nájdennej hranici a označuje pixely na hranici. Po označení celej hranice sa pokračuje v prechode obrazom, až kým sa nedosiahne pravý dolný roh obrázku.

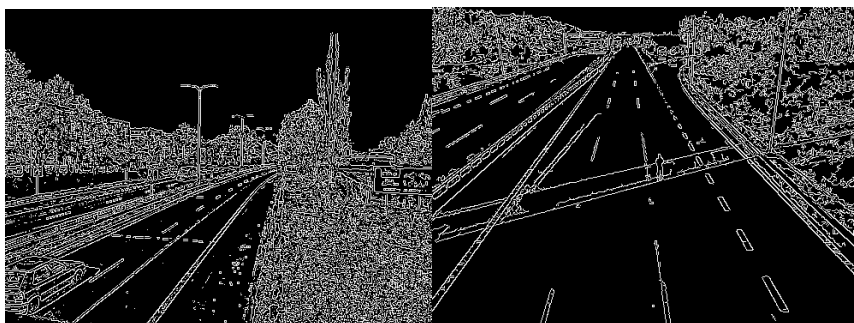
Hľadanie kontúr je najpresnejšie za použitia binárneho obrazu. Výstupom sú súradnice všetkých pixelov, ktoré patria do danej kontúry. Za vozovku sa berie kontúra s najväčšou plochou s dôrazom na ošetrenie situácie, že obloha má väčšiu plochu ako vozovka (ak sa určité percento pixelov kontúry nachádza v hornej časti obrazu, kontúra nie je považovaná za vozovku). Iné časti ako obloha už nemali väčšiu plochu ako vozovka.

Graficky znázornené výstupy sú zobrazené na obrázkoch 6.6. Už na prvý pohľad vidno viditeľné zlepšenie v porovnaní s metódou využívajúcou Houghovu transformáciu. Obrázok na ľavej strane patrí do datasetu stavu vozovky a obrázok vpravo možno zaradiť ako hraničný obrázok z datasetu počasie s dopravnou scénou, v pozadí vidno aspoň trošku oblohu a zároveň je zobrazená aj oblasť mimo vozovky.

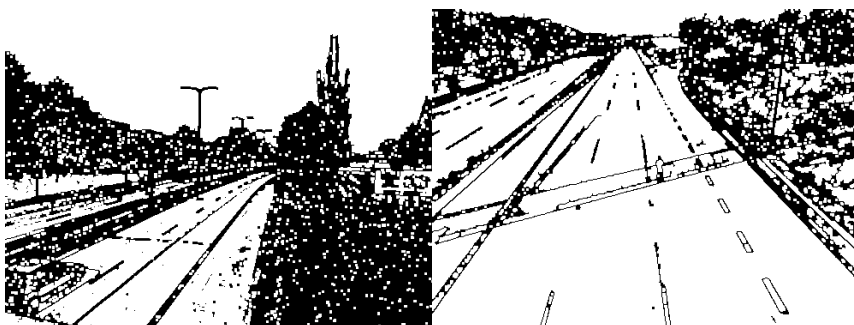
Prvý obrázok je zaujímavý tým, že vozovka je snímaná pod uhlom a nie priamo.



(a) Pôvodný obrázok v RGB



(b) Výstup z Canny detektoru ako



(c) Binárny obrázok po morfológických operáciách a negácii



(d) Výstup algoritmu detekcie vozovky Canny detektorom

Obr. 6.6: Ukážka metódy segmentácie s použitím Canny detektoru

Dopravné kamery sú uložené na rôznych stĺpoch, krajoch budov, rôznych podnosoch a podobne. Tento obrázok znázorňuje polohu kamery, ktorá sa v úlohách snímania dopravy bude vyskytovať pravdepodobne najčastejšie. Metóda založená na Cannyho detektore nie je závislá od pozície kamery a uhla snímania vozovky. Druhou zaujímavosťou na tomto obrázku, ktorá bude v praxi úplne bežná, je prítomnosť vozidla na vozovke. Z výsledku segmentácie opäť vidno, že aj toto dokáže tento algoritmus pomerne dobre zvládnuť. Poslednou zaujímavosťou tohto obrázku je porovnanie plochy kontúry (obrázok c)) vozovky a oblasti mimo vozovky, konkrétne oblohy. Tento problém som vyriešil tak, že okrem porovnávania plochy kontúry beriem do úvahy aj pozíciu pixelov. Ak sa nachádza určité percento (hodnota v tomto prípade je 20%) v hornej polovici obrázka, túto kontúru ignorujem.

Druhý obrázok je taktiež dobrou ukážkou, lebo výrazná časť zobrazenej vozovky sa nachádza v tieni a zvyšná časť je osvetlená Slnkom. Rozdelenie vozovky na osvetlenú a neosvetlenú je bežným stavom v reálnej doprave. Takáto situácia sa dá vyriešiť zadáním vysokých hodnôt prahov hysterézneho prahovania v Cannyho detektore, konkrétne použité hodnoty prahov pre túto scénu sú 91 a 92 pre nižší prah a vyšší prah. Pre dosiahnutie čo najlepších výsledkov je potrebné malé doladenie pre každú scénu.

Z výsledkov segmentácie možno konštatovať, že aj pomerne jednoduchá metóda dokáže vyriešiť bežné problémy, ktoré sa môžu vyskytnúť v reálnych aplikáciach a zároveň podať spoľahlivé výsledky.

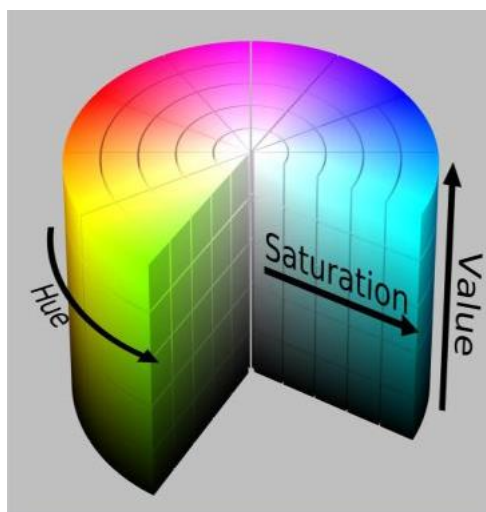
Segmentácia vozovky - metóda HSV modelu

Posledná metóda, ktorú som vyskúšal v rámci segmentácie vozovky. Predchádzajúce dve metódy sú založené na "pozícii" vozovky a jej hraníc s okolitou scénou. Metóda využívajúca HSV model sa pozerá na vozovku z farebného hľadiska, čiže ako "vyzerá". Asfalt má iné farebné zložky ako obrubníky alebo chodníky, lúka má iné rozloženie farieb a podobne. Hlavnou myšlienkou tejto metódy je oddelenie vozovky od okolitých oblastí na základe farebnej rozdielnosti.

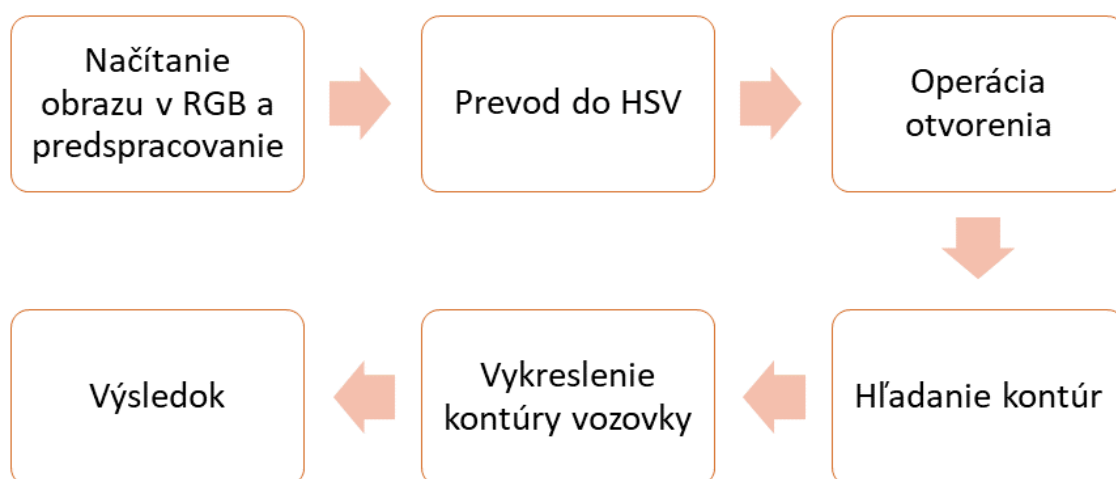
Výhodou HSV modelu je oddelenie informácie o farbe od informácie o intenzite. Je to farebný model, ktorý odpovedá ľudskému vnímaniu farieb. Je tvorený tromi kanálmi a ich zmenou dostávame rôzne kombinácie farieb.

1. **Hue (odtieň)** je farebná časť modelu vyjadrená číslom od 0 do 360 stupňov. Odtieň reprezentuje uhol s vertikálnou osou.
2. **Saturation (sýtosť)** popisuje pomer čistoty odtieňa - množstvo bielej farby zmiešanej s odtieňom. Môže nadobudnúť hodnoty od 0 do 1.
3. **Value (hodnota)** funguje v spojení so sýtosťou a popisuje jas alebo intenzitu farby v rozsahu od 0 do 1, kde 0 je úplne čierna a 1 je najjasnejšia a odhaľuje

najviac farieb. Čím je hodnota nižšia, tým je podobnejšia čiernej a čím je hodnota vyššia, tým je podobnejšia samotnej farbe.



Obr. 6.7: HSV farebný model [52]



Obr. 6.8: Postup algoritmu segmentácie založenej na HSV modeli

Ako prvý krok je opäť načítanie vstupného obrazu a jeho zmenšenie so zanechaním pomeru veľkosti strán pôvodného obrazu na hraničnú hodnotu 500 pixelov.

Druhým krokom je prevod vstupu do farebného modelu HSV a rozdelenie na jednotlivé kanály H, S a V. Po zvolení vhodných parametrov jednotlivých kanálov pomocou trackbaru je výstup vo forme binárneho obrazu.

Ďalším krokom algoritmu je krok morfolologickej operácie otvorenia pre vylepšenie výsledku pred hľadaním kontúr. Pre rýchle pripomenutie, otvorenie sa skladá z operácie erózie, ktorej sekunduje operácia dilatácie. Výsledkom otvorenia je odstránenie šumu a teda lepšia izolácia oblasti vozovky od okolia.

Poslednými krokmi sú vyhľadávanie kontúr a ich vykreslenie. Ako bolo popísané pri predchádzajúcej metóde, výstupom sú indexy pixelov, ktoré patria do danej kontúry vozovky. Prevedenie tohto kroku je totožné ako pri segmentácii s použitím Canny detektoru.

Grafické výstupy sú znázornené na obrázku 6.9. Sú zobrazené výsledky segmentácie použitím všetkých kanálov H, S a V súčasne (prvý stĺpec) a následne vykreslenie výstupov segmentácie práve jedného kanálu - druhý stĺpec odpovedá kanálu H, tretí stĺpec kanálu S a posledný kanálu V.

V prípade použitia všetkých troch kanálov je nutné nastaviť spodnú a hornú hranicu každého kanálu zvlášť, čo činí šesť parametrov. V ostatných prípadoch stačí nastaviť dva parametre. Z výsledkov segmentácie vidno, že použitie samostatného kanálu saturácie S nedáva uspokojivé výsledky. Kanál H má o niečo lepšie výsledky avšak v prípadoch so zamračenou oblohou je náročné oddeliť vozovku od oblohy, kvôli ich podobnej farbe (prvý obrázok stĺpec 2). Samostatný kanál V dáva veľmi podobné výsledky, ako použitie všetkých kanálov spolu s tou výhodou, že namiesto šiestich parametrov stačí nastavovať iba dva, čo je určite efektívnejšie riešenie. Z tohto dôvodu som pri použití HSV modelu na segmentáciu používal iba kanál V.

Analýza stavu vozovky

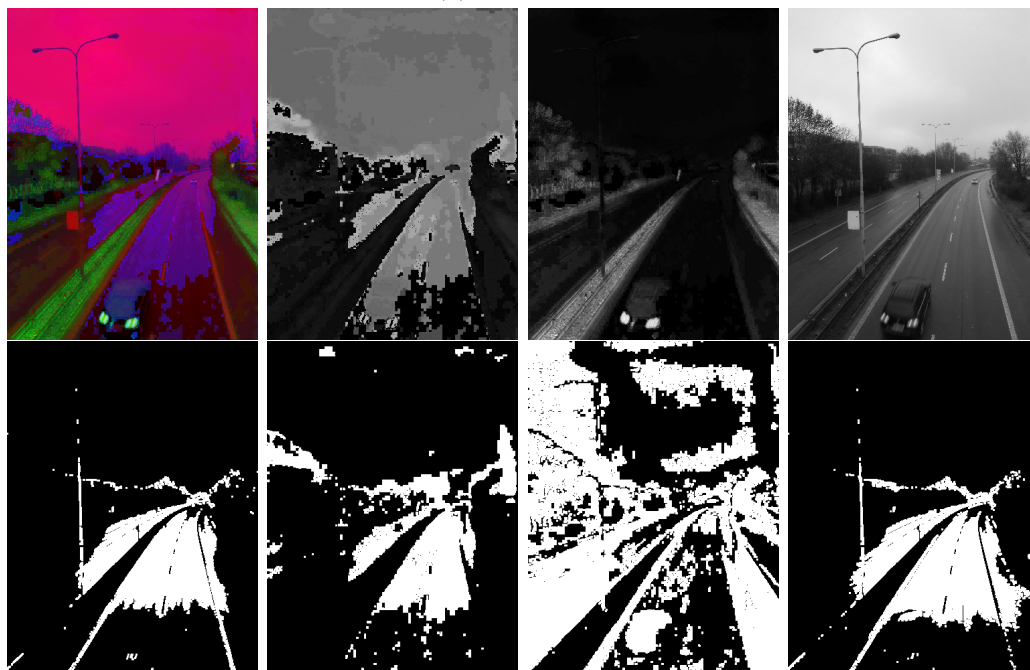
Po úspešnom oddelení vozovky od zvyšku obrazu môže nastať analýza vozovky. Pod pojmom analýza vozovky si možno predstaviť určenie jej stavu - či je suchá, mokrá alebo zasnežená. Ako som už spomenul v rozbere na začiatku kapitoly, implementoval som jednu metódu, ktorá je založená na histograme a skúma rozloženie farieb v oblasti vozovky.

Histogram obrazu je grafické znázornenie závislosti početnosti jasových úrovní obrazu na hodnotách týchto úrovní. Inak povedané, pomocou histogramu môžeme zistiť, koľko pixelov v obraze má konkrétnu hodnotu intenzity (napr. 100). Vodorovná os zobrazuje jasové úrovne (intenzitu pixelov) a vertikálna os zobrazuje počet pixelov. Histogram zo šedotónového obrazu (1D histogram) poskytuje globálnu informáciu o danom obraze, nejde z neho určiť, aké objekty alebo aké vlastnosti má daný obraz. V úlohe analýzy stavu vozovky však nie je potrebné hľadať špecifické objekty alebo vlastnosti, dôležité je iba rozloženie farieb.

Vstupom algoritmu je vysegmentovaná časť vozovky ako šedotónový obraz, z ktorého je vypočítaný a vykreslený histogram. Druhá časť riešenia postupne prechádza všetkými pixelmi a rozdelí ich podľa hodnoty intenzity do troch tried - suchý, mokrý a zasnežený. Na rozdelenie je potrebná určitá referenčná hodnota, podľa ktorej sa bude vyberať príslušná trieda. Táto kapitola popisuje hlavné myšlienky a spôsob fungovania algoritmov ako samostatných jednotiek, upresnenie spôsobu fungovania



(a) RGB vstup



(b) Všetky kanály

(c) H kanál

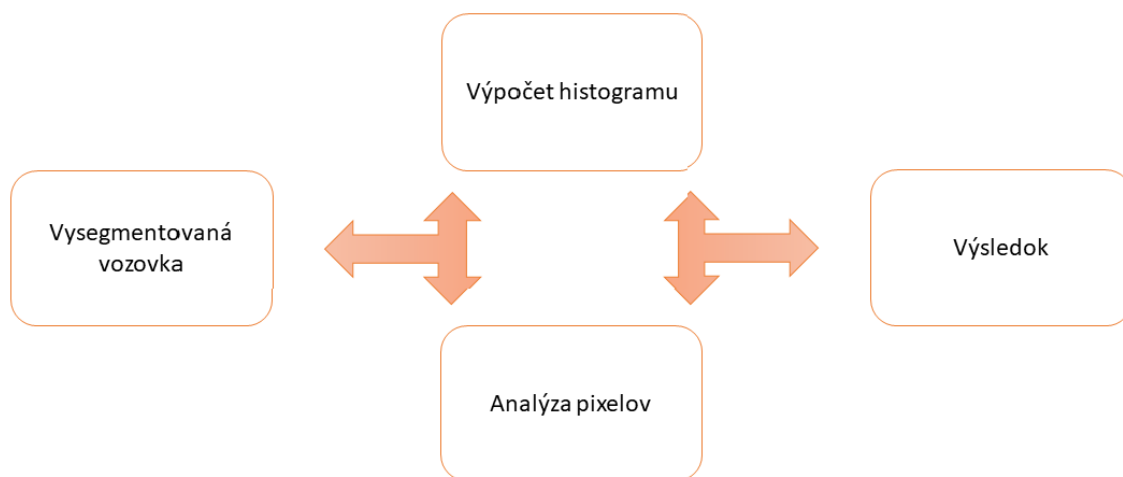
(d) S kanál

(e) V kanál

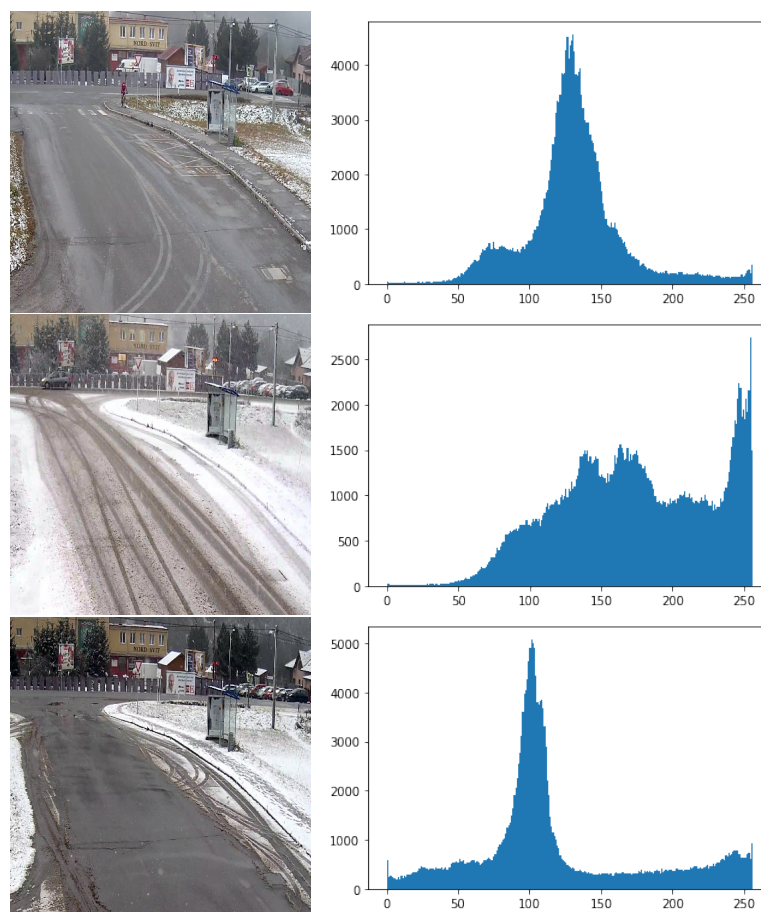
Obr. 6.9: Porovnanie výsledkou segmentácie vozovky použitím všetkých kanálov HSV s ich samotným použitím

bude popísané v ďalšej kapitole.

Z grafických výstupov zobrazených na obrázkoch 6.11 a 6.12 je vidno, že aj takéto jednoduché riešenie, ako histogram zo šedotónového obrázku, môže mať uspokojivé

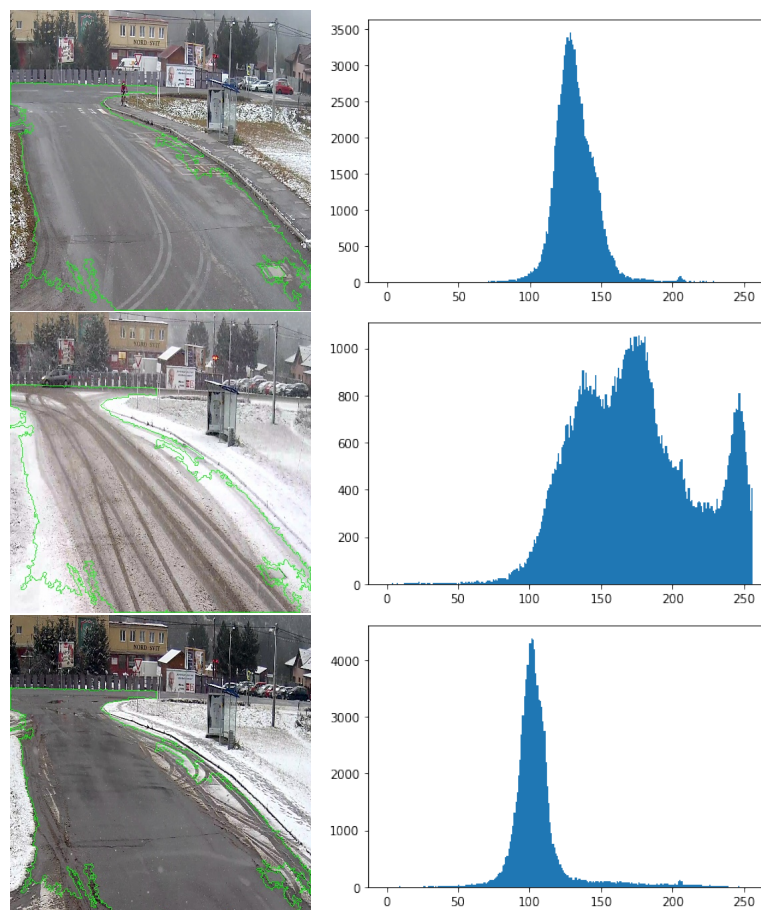


Obr. 6.10: Postup algoritmu analýzy vozovky



Obr. 6.11: Porovnanie histogramov z celého obrázku na suchú, zasneženú a mokrú vozovku. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov.

výsledky. Napriek tomu, že histogramy na 6.11 sú vykreslené z obsahu celých obrázkov, dá sa z nich, v tomto konkrétnom prípade, na hrubo určiť správny stav vozovky.



Obr. 6.12: Porovnanie histogramov z vysegmentovanej oblasti suchej, zasneženej a mokrej vozovky. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov.

Toto je špecifický príklad, ktorý znázorňuje, ako vyzerá scéna, ktorá nezachytáva oblohu. Dôvodom je umiestnenie a natočenie kamery na vozovku. Vďaka absencii oblohy sa percentuálna oblasť z celkovej plochy obrazu, na ktorej sa vyskytuje vozovka, zväčšuje, čo má za následok správne určenie stavu vozovky z histogramu celého obrázku.

Pokus o určenie stavu vozovky z celého obrázku nie je spoľahlivý ani robustný, a pravdepodobne vo väčšine reálnych prípadov zlyhá. Porovnanie presnosti určenia stavu vozovky z celého obrázku a len z oblasti vozovky možno vidieť na vstupnom obrázku č. 2. Kým v prvom prípade má histogram maximum takmer na hornej hranici (intenzita s hodnotou 255) čo možno s určitosťou považovať za farbu snehu, druhý histogram na obrázku 6.12 má maximum v intenzite približne 175, čo skôr odpovedá farbe mokrého/kašovitého snehu, ako je zobrazený na vstupnom obrázku.

6.2.2 Detekcia zrážok

Ako som spomenul v úvode kapitoly, zrážky si možno predstaviť ako malé objekty s rovnakou farbou, podobným tvarom a smerom pohybu. Ako možný prístup detekcie zrážok sa javí analýza konkrétnej časti obrazu a pozorovanie farebných zmien lokálnych oblastí v obraze - na tomto princípe funguje analýza stavu vozovky, ktorá nepriamo detekuje aj prítomnosť zrážok. Druhou možnosťou je hľadanie malých, farebne, tvarovo a orientovane podobných objektov. Snehovú vločku si teda možno predstaviť ako spojenú skupinu pixelov, ktoré zdieľajú spoločnú vlastnosť, napr. intenzita v šedotónovom obrázku. Takáto skupina sa označuje ako blob. Ako ďalšia možnosť detekcie zrážok je detekcia pohybu zrážok v scéne. Všetky tri uvedené možnosti boli implementované a vyskúšané (analýza vozovky, SimpleBlobDetector, dynamický model prostredia).

Detekcia zrážok - metóda hľadania podobných objektov

Pre účely metódy bol použitý SimpleBlobDetector, ktorý je súčasťou OpenCV, ktorý sa skladá z viacerých jednoduchších algoritmov.

Prvým krokom je prahovanie. Úlohou prahovania je previesť vstupný obrázok do niekoľkých binárnych obrázkov použitím rôznych hodnôt prahu. Prahovanie začína od počiatočnej minimálnej hodnoty a postupne sa hodnota prahu zväčšuje pričítaním kroku až na koncovú hodnotu.

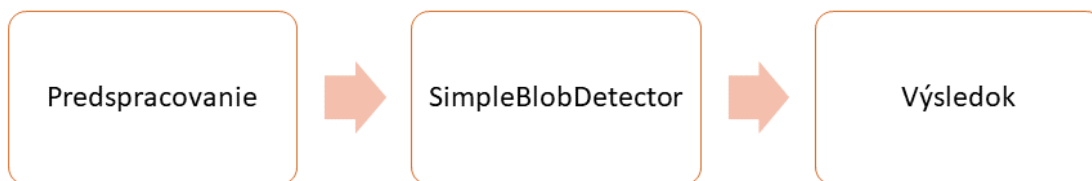
Druhým krokom je extrahovanie spojených oblastí z každého binárneho obrazu pomocou kontúr a následné vypočítanie ich stredov. Výsledkom sú binárne bloby.

V treťom kroku sú vypočítané stredy binárnych blobov v binárnych obrazoch a bloby, ktorých stredy sa nachádzajú v určitom intervale vzdialenosti budú zlúčené. V poslednom kroku sú vypočítané stredy a polomery novospojených blobov.

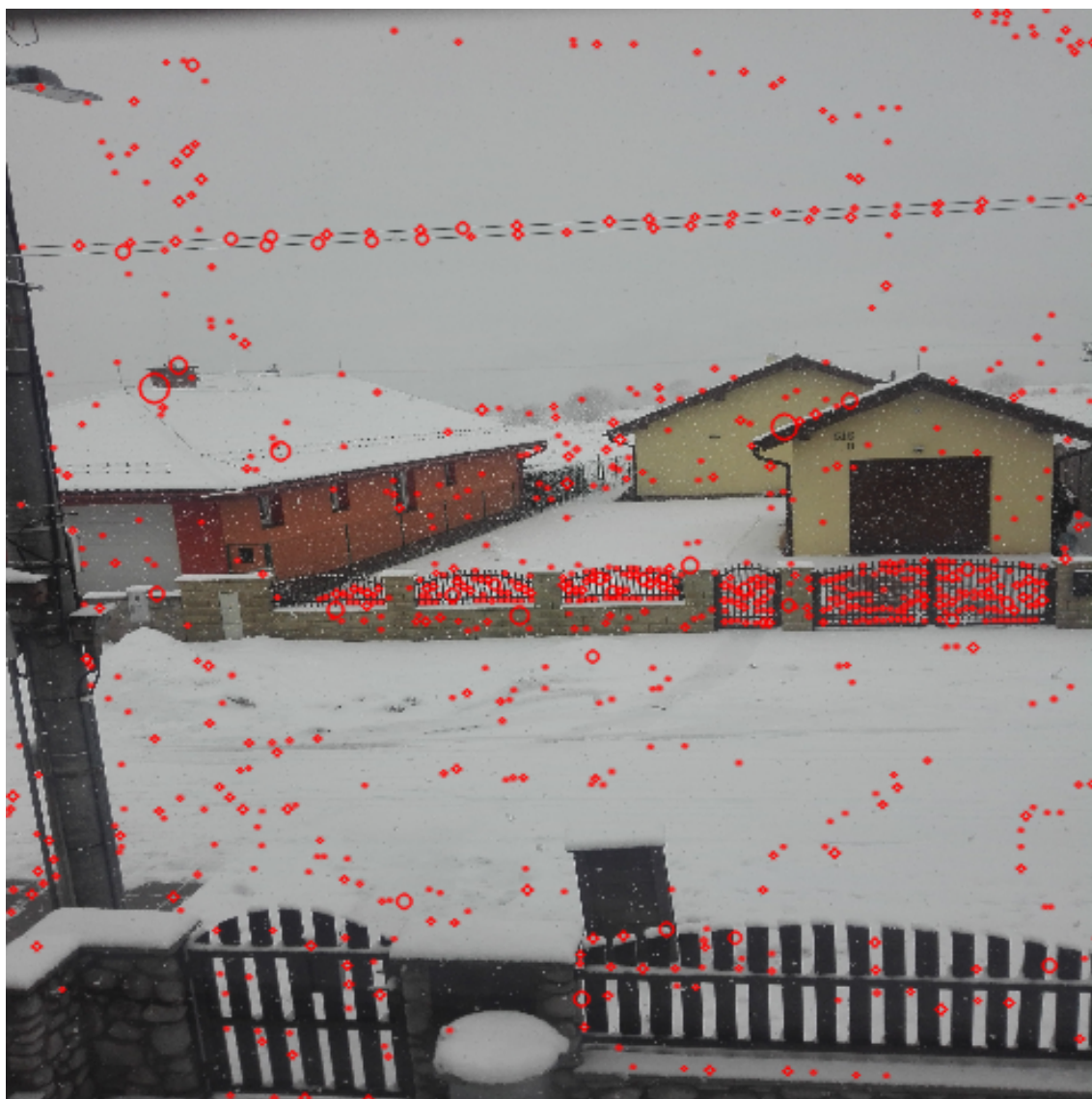
SimpleBlobDetector umožňuje filtrovať bloby na základe farby, veľkosti a tvaru. Filtrovanie na základe farby funguje tak, že filter porovnáva intenzitu v strede blobu s intenzitou blobColor (nastaviteľný parameter). Ak zhoda nenastane, blob je odfiltrovaný. Pre detekciu tmavých blobov, je potrebné nastaviť blobColor na hodnotu 0, pre svetlé bloby na hodnotu 255.

Filtrácia na základe veľkosti ponechá tie bloby, ktoré majú veľkosť v rozmedzí zadaného intervalu (minArea , maxArea). Filtrácia na základe tvaru má tri parametre a to kruhovitost', konvexnosť a inerciu.

Z grafického výsledku metódy zobrazeného na 6.14 je na prvý pohľad vidno, že táto metóda nie je dostatočne efektívna, vyskytuje sa tam pomerne veľa falošných detekcií, veľa vločiek nie je detekovaných vôbec, čo znižuje úspešnosť detekcie. Vychádzal som hlavne z detekcie podľa veľkosti a podľa farby. Na základe uvedených nedostatkov, túto metódu považujem v úlohe detekcie zrážok za nevhodnú.



Obr. 6.13: Postup algoritmu hľadania blobov

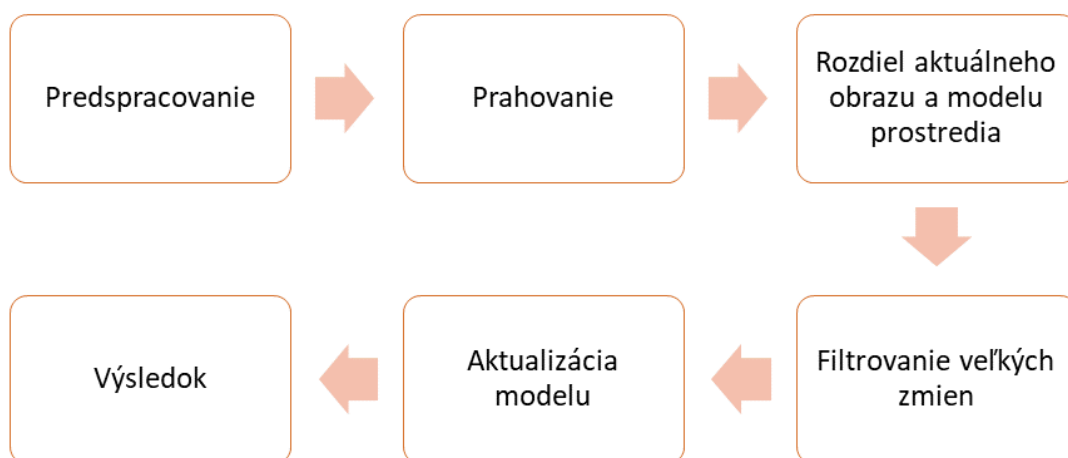


Obr. 6.14: Výsledok použitia SimpleBlobDetector

Detekcia zrážok - metóda dynamického modelu

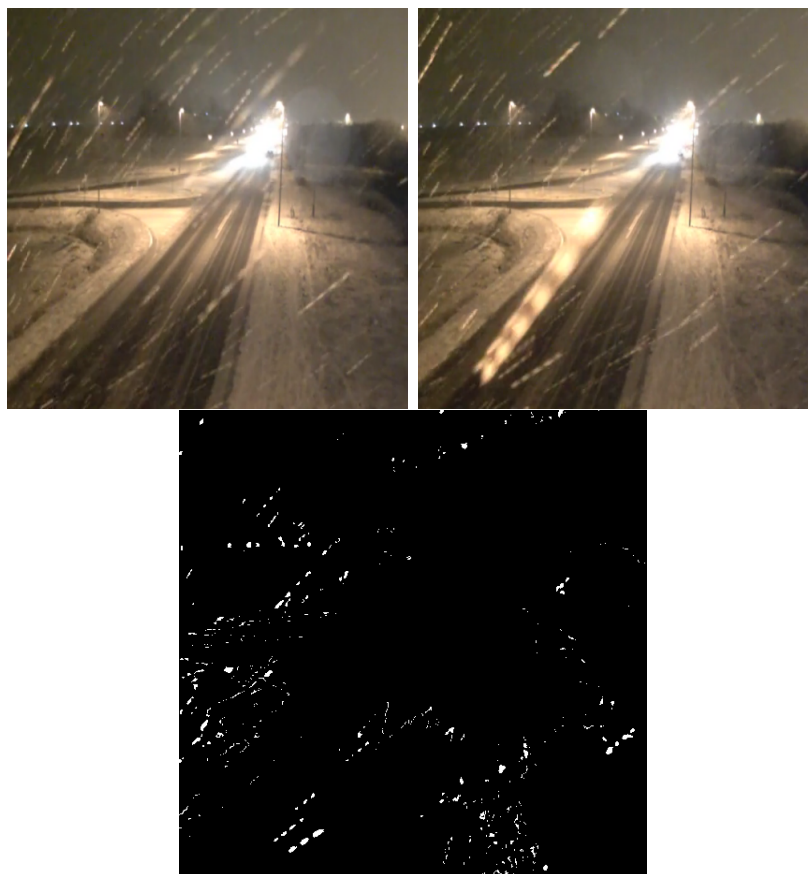
Táto metóda vychádza z toho, že zrážky sa v scéne pohybujú v rovnakom smere. Výsledkom pohybu sú zmeny pozícií jednotlivých zrážok v obraze, ktoré je možné

detekovať. Na základe veľkosti zmien možno jednak detekovať prítomnosť zrážok, ale aj určiť ich intenzitu. Princíp metódy spočíva vo vytvorení dynamického modelu prostredia, ktorý je dynamicky aktualizovaný.



Obr. 6.15: Postup algoritmu detekcie zrážok

Prvý krok algoritmu je predspracovanie, ktoré prevedie vstupný RGB obraz na šedotónový. Druhým krokom je prahovanie pre zvýraznenie zrážok v obraze. Ďalším krokom je rozdiel aktuálne analyzovaného obrazu a modelu prostredia. Oba obrazy sú binárne, takže pixely s hodnotou 1 by mali byť len tie, pri ktorých nastala nejaká zmena pozície. Keďže sa jedná o scény z dopravy, je treba počítať aj s pohybom chodcov, automobilov a podobne. V prípade ignorovania pohybu týchto, podstatne väčších objektov ako sú zrážky, dosiahnuté výsledky by boli veľmi skreslené a nepresné. Z toho dôvodu je ďalším krokom algoritmu hľadanie kontúr. Kontúra bude odfiltrovaná, ak je oblasť, ktorú ohraničuje väčšia ako zadaná prahová hodnota (napr. 5 pixelov, všetkým pixelom patriacim do danej kontúry sú prepísané hodnoty na 0). Na výpočet počtu pixelov, ktoré obklopuje kontúra, bola použitá funkcia z OpenCV `contourArea()`. Týmto spôsobom sa eliminuje detekcia nežiadúceho pohybu v obraze. Pre dosiahnutie čo najlepších výsledkov, je nutná pravidelná aktualizácia modelu prostredia, aby prirodzené vplyvy ako napr. *"pohyb"* tieňa počas dňa neovplyvňovali presnosť výsledku. Pri testovaní metódy som pozadie aktualizoval po každom cykle. Výsledkom algoritmu je počet pixelov, ktorým sa zmenila pozícia vplyvom pohybu. Vo výsledku je prítomný aj šum, lebo je použitá len jemná filtrácia s jadrom 1x1, aby nebolo odfiltrované aj veľké množstvo zrážok, čo by zhoršilo detekciu. Prítomnosť šumu môže byť zanedbaná, dôležitým výsledkom je, či nastala a aká veľká zmena nastala.



Obr. 6.16: Výsledok metódy založenej na dynamickom modeli prostredia

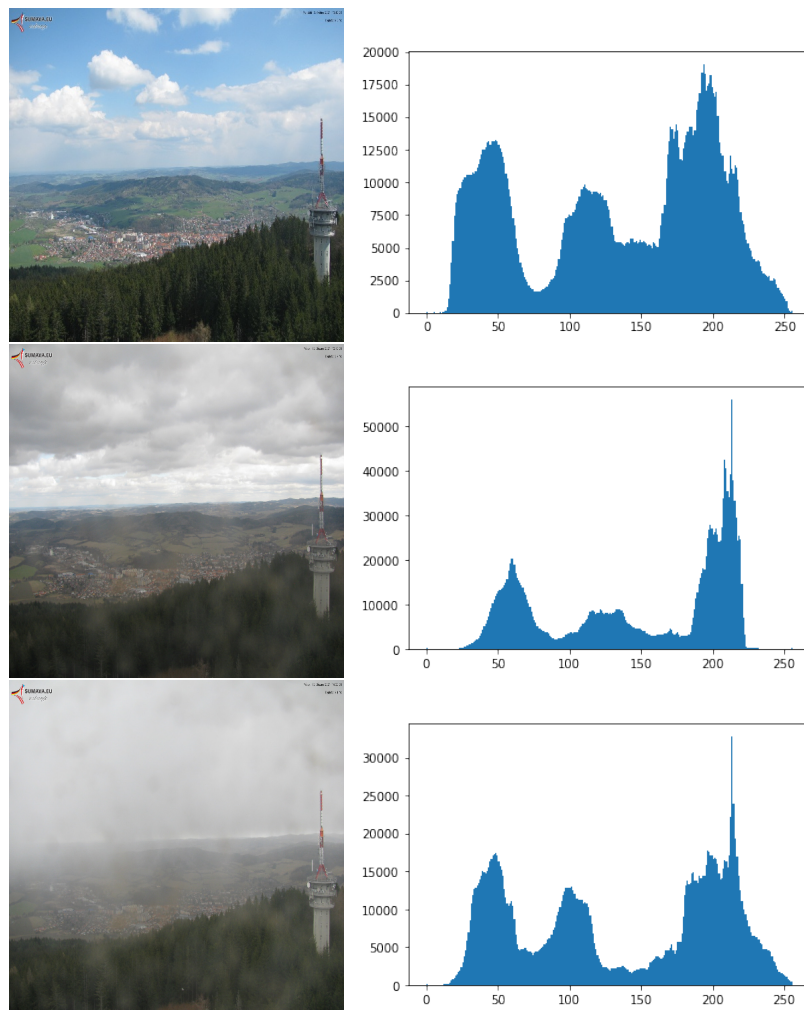
6.2.3 Detekcia hmly

Detekcia prítomnosti hmly je zložitejšia úloha ako analýza stavu vozovky hlavne z toho dôvodu, že teraz už nie je na obraze pevne daná oblasť, ktorú stačí analyzovať. V prípade vozovky stačilo analyzovať lokálnu oblasť (vysegmentovaná vozovka), v tomto prípade je riešenie podstatne závislejšie na konkrétnej scéne obrazu - pozadie bez oblohy, pozadie s nízkym počtom objektov a podobne.

Detekcia hmly - metóda histogramu

Prvá metóda, ktorou som chcel detekovať prítomnosť hmly na obraze je metóda založená na histograme. Princíp je rovnaký ako pri analýze stavu vozovky - vstupom algoritmu je šedotónový obraz, z ktorého je vypočítaný histogram. Táto metóda je dosť všeobecná a spoľahlivo dokáže rozlíšiť iba výrazne odlišné stavy počasia (slnečno od zamračeného alebo slnečno od hmly) a nedokáže spoľahlivo odlíšiť podobné stavy počasia (zamračené od hmly).

Z grafického výstupu je možné vidieť, že táto metóda je vhodná na pomerne jednoduché odlíšenie slnečného počasia od zamračeného resp. hmly. V prípade odlíšenia

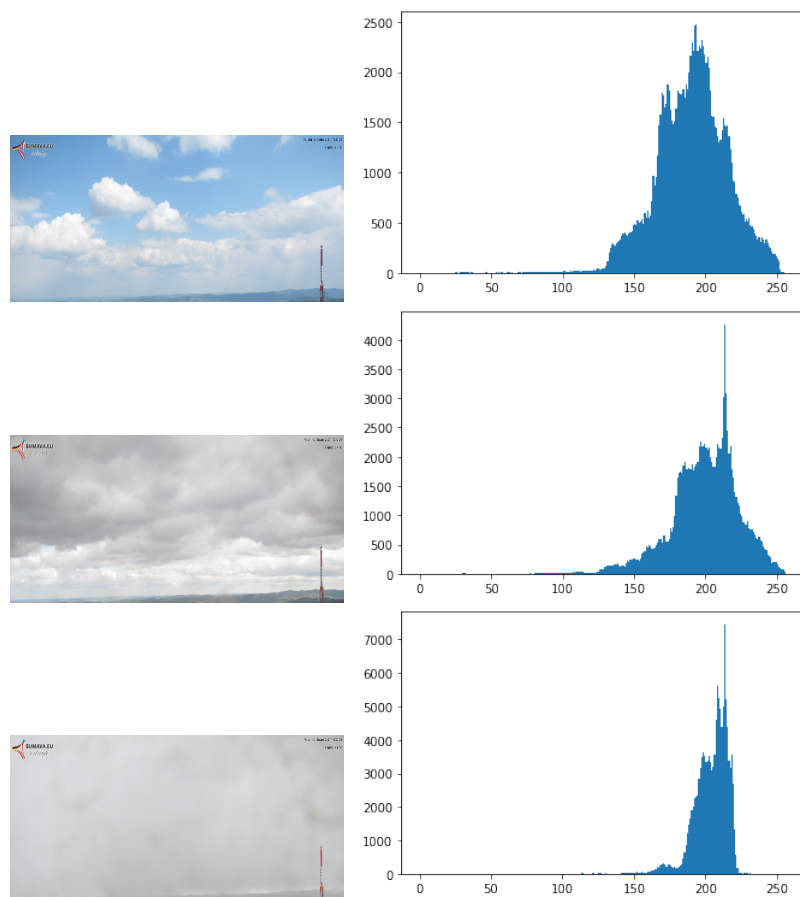


Obr. 6.17: Porovnanie histogramov slnečného, zamračeného a hmlistého počasia. Histogramy sú vykreslené zo šedotónového variantu zobrazených obrázkov [51].

hmly od zamračeného sa táto metóda nejaví ako dostačujúce riešenie. Farba oblakov na prostrednom obrázku je veľmi podobná s farbou hmly, čo môže viesť k falošnému výsledku klasifikácie, zamračené počasia sa určí ako potvrdenie prítomnosti hmly alebo opačne, prítomnosť hmly sa nesprávne určí ako zamračené počasia. Z toho dôvodu som implementoval ďalšie metódy detekcie hmly pre zlepšenie spoľahlivosti a presnosti detekcie. Obrázok 6.18 zobrazuje porovnanie histogramov iba z hornej polovice vstupných obrazov. Výsledok je o niečo lepší, lebo zobrazuje rozloženie farieb iba v najdôležitejšej oblasti pre detekciu hmly, ale aj tak nepomáha k odlíšeniu hmly od zamračeného počasia.

Detekcia hmly - metóda Cannyho detektoru

Cieľom druhej metódy detekcie hmly je presnejšie rozlíšenie hmly od zamračeného počasia. Hlavným cieľom tejto metódy je analýza počtu hrán v obraze a je založená



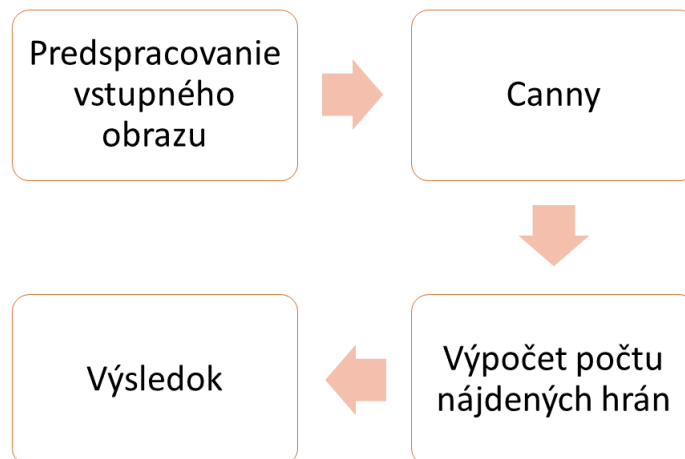
Obr. 6.18: Porovnanie histogramov slnečného, zamračeného a hmlistého počasia. Histogramy sú vykreslené zo šedotónového variantu hornej polovice zobrazených obrázkov.

na myšlienke, že obraz hmlistého počasia by mal mať podstatne menej hrán ako obraz zamračeného ale aj slnečného počasia.

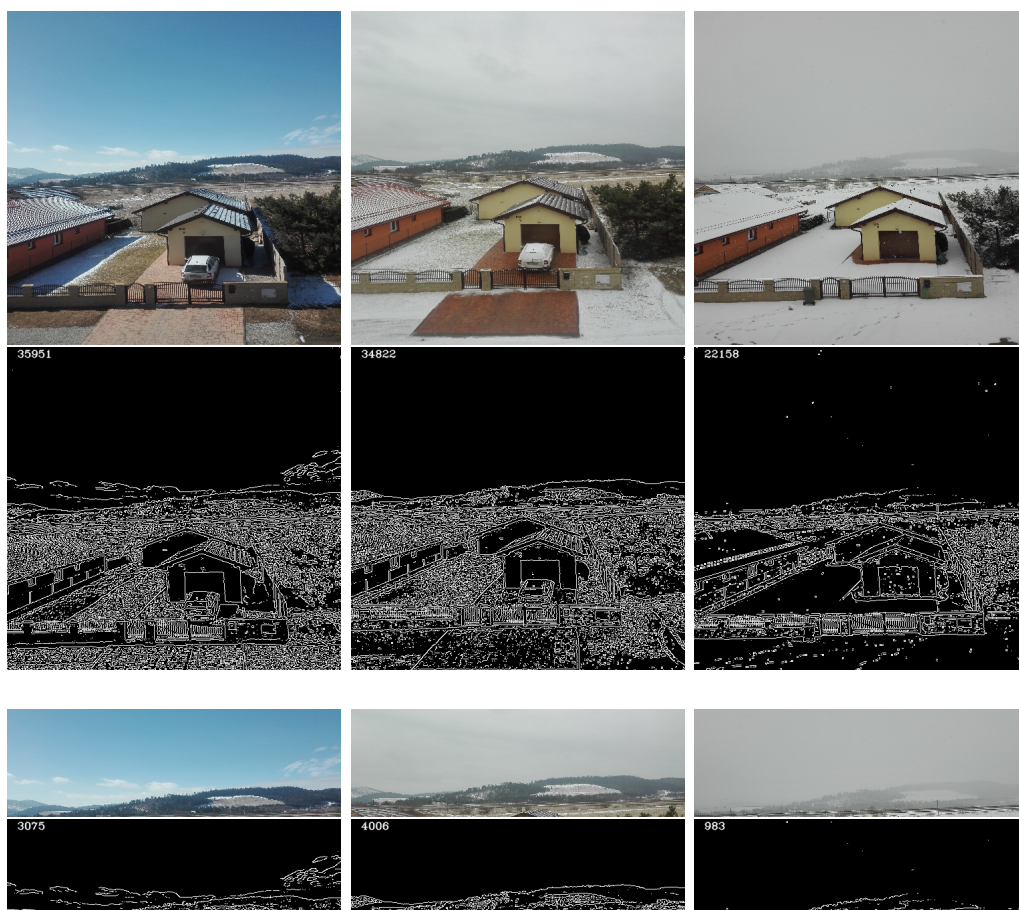
Prvým krokom algoritmu je predspracovanie vstupu - výber hornej časti obrazu, ktorá sa bude spracovávať. Zmenšený obraz je vstupom do Cannyho detektoru hrán, ktorý ho prevedie na binárny obrázok. Posledný krok algoritmu je určenie, koľko bielych pixelov sa vo výstupnom binárnom obraze nachádza.

Grafické výsledky metódy sú zobrazené na obrázkoch 6.20. Opäť som zvolil spracovanie jednak celého obrazu, ale aj jeho hornej časti pre porovnanie výsledkov. Vo všetkých prípadoch boli zvolené rovnaké hodnoty spodných a horných prahov, konkrétne hodnoty 51 a 51. Hodnoty v prípade obrázku sú vybrané manuálne. V prípade reálneho použitia, sú hodnoty prahov určené automaticky použitím Otsuovej metódy, ktorá vypočíta jednu hodnotu prahu, ktorá je použitá ako horný prah a jej polovica ako spodný prah.

Obrazy nie sú rovnako vycentrované lebo som ich fotil "rukou" bez použitia



Obr. 6.19: Postup algoritmu detekcie hmly počítaním hrán



Obr. 6.20: Porovnanie hrán slnečného (35951 resp. 3075), zamračeného (34822 resp. 4006) a hmlistého (22158 resp. 983) počasia

statívu. Tieto drobné odchýlky však v tomto prípade môžu byť zanedbané. Jediné, čo je pre úlohu detekcie hmly podstatné a dôležité, je pozadie a vzdialenejšie objekty

zachytené väčšinou v hornej časti obrázkov. Z toho dôvodu je postačujúce analyzovať len určitú časť obrazu. Tým sa znížia výpočtové zdroje a hlavne sa zvýši presnosť detekcie.

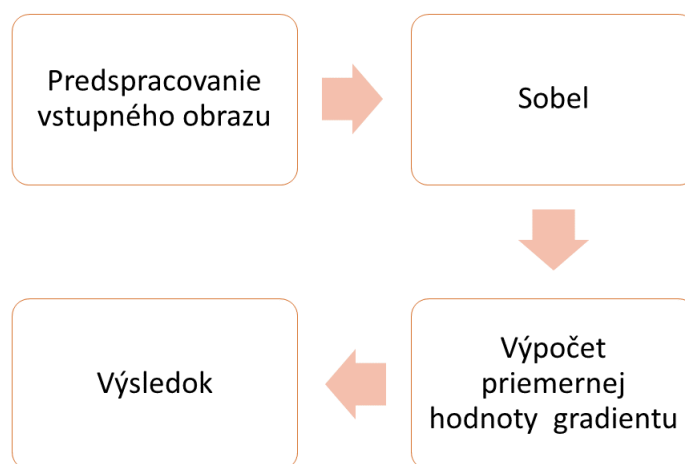
Z výsledkov detekcie hrán v 6.20 v druhom riadku je vidno, že je zbytočné hľadať hrany v celom obraze. Ako vhodné prípady použitia celého obrazu môžu byť stanoviská, ktorých snímacia kamera zvierá veľký uhol s vozovkou alebo miesta, ktoré nemajú v pozadí žiadne objekty, ktorých hrany by boli použité na detekciu.

Posledný vstupný obrázok v 6.20 zachytáva pomerne slabú hmlu, ale aj napriek tomu potvrdzuje, že pomocou hrán v hornej časti obrazu, je možné rozlíšiť hmlisté počasie od zamračeného a do určitej miery aj slnečné od zamračeného počasia.

Detekcia hmly - metóda analýzy kontrastu

Táto metóda je založená na myšlienke, že hmla rozmazáva hrany a tým znižuje kontrast v obraze. Predpokladal som, že obrázok s hmlou bude mať oveľa nižší kontrast ako obrázok zamračeného počasia a ten bude mať vyšší kontrast ako obrázok slnečného počasia.

Prvým krokom algoritmu je predspracovanie vstupu - výber hornej časti obrazu, ktorá sa bude spracovávať. Zmenšený obraz je vstupom do Sobelovho operátora, ktorý vypočítava gradient jasovej funkcie. Na záver algoritmu je vypočítaná priemerná hodnota gradientu jasovej funkcie (pre malý kontrast v obraze je aj priemerný gradient jasovej funkcie malý).



Obr. 6.21: Postup algoritmu analýzy kontrastu

Grafické výstupy metódy zobrazujú porovnanie chovania gradientu v obrázkoch so slnečným, zamračeným a hmlistým počasím. Opäť som porovnal výstup z celého

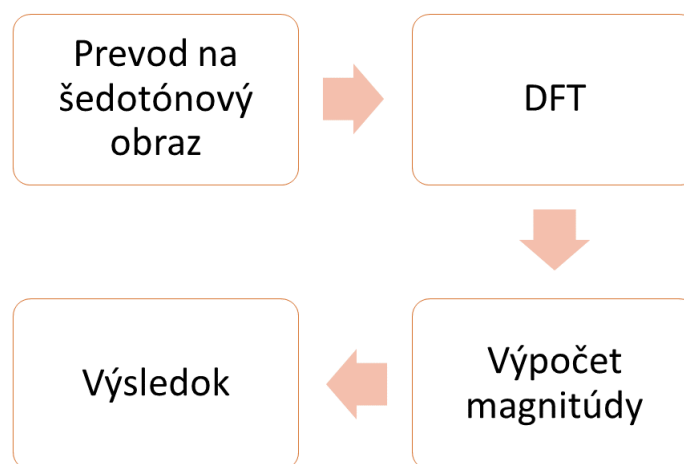


Obr. 6.22: Porovnanie gradientu slnečného (19,02 resp. 42,3), zamračeného (29,54 resp. 68,38) a hmlistého (19,57 resp. 21,37) počasia použitím Sobelovho operátoru

obrázku s výstupom len z hornej polovice obrázku. Výstupné hodnoty gradientov boli určené absolútnymi hodnotami 22, 27 a 20 pre prípad celých obrázkov a 49, 67, 40 pre polovičné varianty, čím sa potvrdil predpoklad, že hmla znižuje kontrast v obraze a je možné ju detekovať analýzou zmeny priemernej hodnoty gradientu jasovej funkcie.

Detekcia hmly - metóda analýzy frekvenčného spektra obrazu

Posledný spôsob, ktorým som sa pokúsil detekovať prítomnosť hmly v obraze je analýza obrazu vo frekvenčnej oblasti. Hlavná myšlienka metódy je založená na tom, že hmla rozmazáva obraz, inak povedané, hrany nie sú také ostré. Ako som spomenul v rozbere na úvode kapitoly, rozmazanie možno prirovnať k absencii vysokých frekvencií (časti obrazu s rapídnuou zmenou, napr. hrany). Čím je teda vyššia frekvencia zmeny (napr. zo svetlej na tmavú), tým je zmena rýchlejšia. Pre prevod obrázku z časovej oblasti do frekvenčnej bola použitá Fourierova transformácia.



Obr. 6.23: Postup algoritmu analýzy obrazu vo frekvenčnej oblasti

Fourierova transformácia pracuje s intenzitou signálu v čase (v prípade 2D signálu ako sú obrázky, v priestore) a prevádza ju na intenzitu signálu vo frekvenčnej oblasti. V prípade viac-kanálového obrázku je potrebné prevádzať jednotlivé kanály zvlášť. Z toho dôvodu som ako vstup zvolil šedotónový obrázok.

Druhý krok algoritmu je samotný výpočet diskkrétnej Fourierovej transformácie. Pred samotným výpočtom je dobré pre urýchlenie výpočtu upraviť rozmer vstupnej matice. Najlepšie voľby sú rozmery, ktoré sú násobkom dvojky, trojky alebo päťky,

čo sa dá dosiahnuť doplnením núl na hranice obrázku. Výpočet Fourierovej transformácie rozdelením problému na viacero menších častí je výpočtetne efektívnejšie. Výsledkom transformácie je matica komplexných čísel.

Následne je vypočítaná magnitúda, ktorá je prevedená do logaritmickkej mierky pre umožnenie zobrazenia vysokých intenzít a je normalizovaná na hodnoty v intervale $[0, 1]$. Z matice magnitúdy je ako výstup metódy vypočítaná jej priemerná hodnota.



Obr. 6.24: Porovnanie magnitúdy frekvenčného spektra slnečného, zamračeného a hmlistého počasia

Grafické výsledky metódy založenej na Fourierovej transformácii sú znázornené na 6.24. Pri pohľade na frekvenčné spektrá sú viditeľné rozdiely, čo potvrdzuje predpoklad, že hmla mení obraz aj z frekvenčného hľadiska. Z výstupu je vidno, že opäť je výraznejšia zmena pri obrázku s hmlou pri analýze iba vrchnej časti.

6.3 Zhrnutie klasických metód

V predchádzajúcich kapitolách a sub-kapitolách je popísaný spôsob fungovania samostatných algoritmov. Zo samotného fungovania ale nie je jasné, ako prebieha vyhodnotenie výsledku a určenie záveru algoritmu. Táto kapitola popíše, ako prebieha detekcia počasia. Zhrnutie použitých príznakov pre jednotlivé časti úlohy detekcie sú spísané v tabuľke 6.1.

Tab. 6.1: Zhrnutie charakteristických príznakov pre jednotlivé metódy

Modul	Charakteristické príznaky
Stav vozovky	zmena rozloženia farby, zmena histogramu
Detekcia hmly	pokles kontrastu, pokles počtu hrán, pokles ostrosti hrán, rozmazanie hrán, zmena frekvencie
Detekcia zrážok	zmena rozloženia farby, pokles počtu hrán, rozmazanosť, detekcia pohybu

Keďže je práca zameraná na dopravnú scénu, možno predpokladať, že vo väčšine príkladov použitia v praxi, bude zdroj vstupných dát jedna statická kamera s pevne danými nastaveniami, ktorá sa nepohybuje, neotáča ani nijak nemení svoju polohu, a teda významné objekty v pozadí (budovy, pohoria, les a podobne) sa v scéne nachádzajú vždy na rovnakom mieste. Tieto predpoklady sú pre vyhodnotenie detekcie veľmi dôležité, keďže hlavná myšlienka použitých algoritmov je porovnávanie zmien, či už počtu, rozloženie farby zmeny gradientu a podobne, čiže zachovanie polohy dôležitých objektov je pre správne fungovanie navrhnutých metód kľúčové. Princíp navrhnutých metód spočíva v porovnávaní aktuálneho, analyzovaného obrazu s nejakou referenciou. Pod referenciou si možno predstaviť *ideálny* obraz.

Obrázok 6.25 zobrazuje parametrizáciu viditeľnosti a intenzity zrážok. Odhad viditeľnosti je určený ako pomer súčtov výsledných hodnôt metód (počet hrán, kontrast, magnitúda) aktuálneho a referenčného obrázka a výsledok je hodnota v percentách. Parametrizácia zrážok je určená podľa množstva bielych pixelov po odčítaní obrázku od pozadia.

6.3.1 Obmedzujúce podmienky analýzy stavu vozovky

Pre rýchle pripomenutie, analýza stavu vozovky sa skladá z dvoch krokov - segmentácia a samotná analýza. Najprv sa vysegmentuje oblasť vozovky a následne sa s touto oblasťou pracuje ďalej, kde sa hľadajú a porovnávajú odchýlky. V prípade postupu porovnávania aktuálneho obrazu s referenčným je potrebné segmentovať



Obr. 6.25: Ukážka parametrizácie viditeľnosti a intenzity zražok

vozovku pre každý obraz zvlášť, čo obsahuje určité problémy. Hlavným problémom je rozdiel veľkostí vysegmentovaných častí pre rôzne stavy počasia (tiene, snehové kolaje) s použitím rovnakých parametrov. Tento problém je možné riešiť doplnením rozmerov menšej matice vysegmentovanej vozovky. Tu ale nastáva otázka, aké zvolit hodnoty intenzity *umelo* doplnených elementov matice, či zvolit priemernú hodnotu intenzity celej oblasti, alebo doplniť hodnoty hraničných pixelov kontúry a podobne. Už na prvý pohľad je jasné, že tento spôsob riešenia je výpočetne náročnejší a celková spoľahlivosť a presnosť doplnením aproximačných pixelov klesá.

Ako efektívnejší spôsob na vyriešenie problému som zvolil vysegmentovanie vozovky iba raz, a to práve na ideálnom, resp. referenčnom obrázku, kde je možné zaistiť čo najpresnejšiu segmentáciu vozovky. Referenčný obrázok sa počas algoritmu neaktualizuje. Aktuálny obrázok dostane súradnice pixelov, ktoré tvoria oblasť vozovky (masku vozovky) a následne môže riešiť analýzu tejto oblasti. Výsledkom je, že samotná segmentačná časť sa počíta iba jedenkrát pre referenčný obrázok, súradnice všetkých pixelov oblasti sú predávané medzi analyzovanými obrázkami a počítajú sa len metódy analýzy vozovky. Nevýhodou riešenia je, že ak sa nachádza automobil v aktuálnom obrázku na mieste, ktoré ideálny obrázok určil ako vozovku, berie sa aj automobil za vozovku. Použitie tejto metódy nebude také spoľahlivé v miestach s vysoko-frekventovanou dopravou. V prípade nasadenia do takýchto podmienok, by bolo lepšie rozdeliť oblasť vysegmentovanej vozovky na viaceré malé časti riešené samostatne a výsledok by bol priemerom zo všetkých malých častí. Ďalšou komplikáciou môžu byť malé odchýlky polohy kamery vplyvom vetra - tento problém som na dodaných testovacích obrázkoch z reálnych staníc nepostrehol.

V analýze stavu vozovky sa porovnáva, aká veľká je zhoda medzi histogramom vozovky z ideálneho a aktuálneho obrázka. Výsledná korelácia je uvedená v percentách a čím vyššia hodnota, tým je podobnosť vozoviek vyššia. Pri testovaní som narazil na problém, že histogram je závislý na zmenách osvetlenia, polohy a veľkosti tieňov a podobne. Tento problém vyriešila pomocná metóda, ktorá prechádza každým pixelom vysegmentovanej oblasti a porovnáva jeho intenzitu s príslušným pixelom z referenčného obrázka. Na základe presne určených prahov sú pixely rozdelené do tried suchá, mokrá a zasnežená. Výstupný stav vozovky je určený najpočetnejšou triedou. Podľa percenta zakrytej vozovky (napr. snehom) je možná aj nepriama detekcia zrážok a zároveň ich parametrizácia. Na konečné testovanie úspešnosti analýzy stavu vozovky som zvolil ako referenciu obrázkov so suchou vozovkou a slnečným počasím.

Najväčším problémom je rozlíšenie mokrej nasvietenej vozovky, ktorá sa leskne a tým má vysoké hodnoty intenzity od zasneženej vozovky, ktorá má taktiež vysoké hodnoty intenzity. Tento problém je jednoducho vyriešiteľný úpravou hodnoty prahovania, podľa ktorej je pixel priradený do triedy zasnežená vozovka. Ak tento spôsob nie je možný, druhou možnosťou je kombinácia výstupu z metódy analýzy stavu vozovky a výstupu detekcie zrážok metódou dynamického modelu. Ak je vozovka osvetlená Slnkom, je nízka pravdepodobnosť, že zároveň aj sneží - nenastane detekcia pohybu. Čiže kombináciou výstupov "*zasneženavozovka*" a "*(ne)detekovanýpohyb*" je možné tieto stavy vozovky odlíšiť. Ako posledná možnosť rozlíšenia lesklej a zasneženej vozovky je, že ak sa nachádza na vozovke snehová vrstva, je zvyčajne rovnomerne rozložená na celej oblasti vozovky. V prípade osvetlenia vozovky sa intenzita zvýši iba v nejakej lokálnej oblasti vozovky, lebo vozovka zaberá veľkú časť obrazu. Čiže vo výsledku bude nejaká časť vozovky s veľkými jasovými hodnotami a druhá časť s oveľa nižšími (mokrá vozovka). Táto možnosť je zložitejšia ako prvé dve, ktoré sú dostačujúce vo väčšine prípadov.

6.3.2 Obmedzujúce podmienky detekcie zrážok

Detekcia zrážok metódou dynamického modelu je založená na podobnom princípe ako analýza stavu vozovky. V tomto prípade nie je potrebný žiadny ideálny obrázok, lebo nie je potrebná žiadna segmentácia. Táto metóda je založená na rozdiely aktuálneho obrázka a modelu prostredia. Ako referenciu, v tomto prípade model prostredia som uvažoval vždy predchádzajúci obrázok - model prostredia sa pravidelne aktualizuje. Rovnako ako pri analýze vozovky, aj tu je predpoklad statickej kamery pre dosiahnutie najlepších výsledkov. V prípade výrazných pohybov kamery, napr. vplyvom silného vetra, sú odchýlky väčšie ako zadaná maximálna hodnota plochy kontúry odfiltrované.

V prípade detekcie sneženia resp. snehových vločiek, je táto metóda určite dostačujúca. Metóda dynamického modelu umožňuje určiť aj intenzitu sneženia na základe porovnávania počtu bielych pixelov po odčítaní aktuálneho obrázku od modelu prostredia. Ak sa snehové vločky na obrázku aktuálnom ani na obrázku modelu nenachádzajú, je výsledný počet bielych pixelov v sekvencii obrázkov veľmi podobný. Akonáhle začne snežiť, počty bielych pixelov rastú s intenzitou a veľkosťou snehových vločiek. Intenzita je celé kladné číslo, ktoré sa zvýši po prekonaní stanovenej hodnoty (po pribudnutí napr. 1000 bielych pixelov od posledného rozdielu sa zvýši intenzita o 1 a naopak, ak sa počet pixelov zníži o hodnotu 1000, zníži sa aj intenzita o 1). Intenzita s hodnotou 0 hovorí o neprítomnosti pohybu vplyvom zrážok. Druhou metódou je nepriama detekcia sneženia z analýzy stavu vozovky.

Detekcia dažďa je komplikovanejšia, jednak kvôli vlastnostiam vody, ale aj menšej veľkosti kvapiek v porovnaní so snehovými vločkami. Najväčším problémom je, že kvapky dažďa sú len ťažko viditeľné na obrázkoch s dažďom. Princíp metódy je totožný ako pri detekovaní snehových vločiek. Aj keď nejaké kvapky viditeľné sú, po odčítaní aktuálneho a referenčného obrázku len ťažko odlišiteľné od šumu. Z toho dôvodu je použitie metódy na detekciu pohybu kvapiek dažďa vo väčšine prípadov bez dostačujúcich výsledkov. Jediná metóda detekcie dažďam ktorá ma uspokojivé výsledky, je metóda analýzy stavu vozovky. Okrem detekcie zmien okolia vplyvom dažďa a detekciou pohybujúcich sa kvapiek, už nie je veľmi z čoho ďalšieho vychádzať pri pokuse detekcie dažďa.

6.3.3 Obmedzujúce podmienky detekcie hmly

Detekcia hmly pracuje na rovnakom princípe ako analýza vozovky, porovnávanie referenčného obrázku s aktuálnym obrázkom. Ako referenčný obrázok som zvolil obraz bez hmly, počas algoritmu sa neaktualizuje. Opäť sa využíva výhoda statickej kamery pre dosiahnutie čo najpresnejších výsledkov. Ako bolo ukázané v kapitole popisujúcej navrhnuté riešenia, lepšie výsledky sú dosiahnuté použitím výseku hornej časti obrazu namiesto analýzy celého obrazu - vplyv hmly na obraze sa najlepšie analyzuje na vzdialených objektoch a najvzdialenejšie objekty sa vyskytujú práve v hornej časti obrazu. Zároveň sa skráti čas výpočtu a potrebná pamäť. Nastáva otázka, ako zvoliť čo najlepšiu časť horného obrázku. Opäť, je veľmi náročné zvoliť také parametre, ktoré by pasovali na akúkoľvek dopravnú scénu, sú kamery, ktoré nezachytávajú oblohu, alebo zachytávajú oblohu na väčšine hornej polovice obrázku (nížinatá oblasť) a podobne. Vo finálnom testovaní algoritmov som pracoval práve s obrazmi, ktoré nemali zachytenú oblohu.

Pre čo najpresnejšiu detekciu hmly a zároveň pre možnosť odlíšenia hlavne zamračeného počasia od hmly je dostačujúce skombinovať výstupy z metód skúma-

júcich zmeny počtu hrán a zmeny kontrastu. Obrázok s hmlou má výrazne menej hrán a zároveň nižší kontrast ako obraz zamračeného počasia aj ako obraz slnečného počasia. Zamračené počasia má viac hrán ako slnečné a podobnú hodnotu kontrastu. Rozlíšenie hmly od daždivého počasia resp. sneženia je možné opäť s pomocou metód detekcie zrážok. Hmla je ale zvyčajne doprovdným javom dažďa alebo sneženia. Parametrizácia viditeľnosti je určená ako priemerná hodnota pomerov počtu hrán, hodnoty priemerného kontrastu a hodnoty priemernej magnitúdy aktuálneho a referenčného obrázku. Výsledná odhadnutá hodnota viditeľnosti je v percentách.

6.4 Testovanie algoritmov klasických metód

Pre finálne vyhodnotenie presnosti a spoľahlivosti som použil dataset, ktorý mi bol dodaný školiteľkou práce. Dataset je z reálneho dopravného provozu z troch stanovísk. Kvôli absencii oblohy na všetkých troch stanoviskách je testovanie zamerané na určenie stavu vozovky (suchá, mokrá, zasnežená), binárnej klasifikácii detekcie hmly (prítomná, neprítomná) a detekcii prítomnosti zrážok. Z toho dôvodu, že algoritmy sú postavené na predpoklade rovnakej scény, sú testované jednotlivé stanoviská samostatne. Počty testovacích datasetov boli vytvorené s prihliadnutím k možným rôznym stavom, ktoré môžu na stanoviskách nastať. Nevidel som zmysel v zvolení väčšieho počtu obrazov a testovania stále tých istých stavov za podobných podmienok. Výsledky jednotlivých testovaní sú uložené v textových dokumentoch vrámci programu. Vybrané testovacie obrázky som musel vybrať a anotovať ručne. V prípade anotácie parametrizácie sa jedná o dosť náročnú a subjektívnu úlohu, preto som zameral na binárne testovanie prítomné/neprítomné.

Prvé stanovisko zachytáva rovný úsek vozovky bez zatáčky. Jedná sa o obvyčajnú hlavnú cestu, s dvomi jazdnými pruhmi, ktorá nemá na svojich krajniciach nakreslené biele pruhy. Pozadie je veľmi jednoduché, tvorí ho iba časť lúky. Snímacia kamera je naklonená pod pomerne veľkým uhlom a preto nie je zachytená žiadna časť oblohy.

Snímacia kamera druhého stanoviska je umiestnená na kopci a taktiež je naklonená tak, že nie je zobrazená obloha. Pozadie je v porovnaní so stanoviskom jeden pestrejšie, okrem lúky je snímaná aj časť obytnej oblasti s rodinnými domami. Vozovka obsahuje tri jazdné pruhy (jeden pruh je kvôli stúpaniu rozšírený). Druhým rozdielom v porovnaní so stanoviskom jeden je, že stanovisko dva sníma zakrivenú vozovku (zatáčky). Táto vozovka má na oboch svojich krajniciach nakreslené biele čiary.

Posledné tretie stanovisko je rovnaké ako stanovisko dva s tým rozdielom, že sníma opačný smer - smer klesania z kopca. Samotná kamera je umiestnená v zatáčke pred vrcholom kopca a je zachytená iba veľmi malá časť oblohy. Pozadie nie je také

pestré ako v prípade stanoviska dva lebo obsahuje len minimálne časti domov. V porovnaní s pozadím prvého stanoviska je o niečo pestrejšie, okrem kopcovitého terénu lúky obsahuje aj viac stromov, časť príjazdovej cesty a podobne.

6.4.1 Testovanie stavu vozovky

Vybral som dokopy 389 obrázkov, 135 so suchou, 141 s mokrou a 113 so zasneženou vozovkou. zo všetkých troch stanovísk. Vybral som obrázky s rôznou polohou tieňov, lesknúcu vozovku, so snehovými kolajmi, s rôznou úrovňou snehu na vozovke a podobne, pre zaistenie čo najviac rôznych situácií, ktoré môžu nastať. S použitím rôznych stanovísk možno predpokladať testovanie aj rôznych typov asfaltov. Hodnoty prahov triedenia pixelov boli mierne poupravené pre každé stanovisko, ich hodnoty sú označené v programe. Získané výsledky sú výstupmi iba metód analýzy vozovky. Najväčší problém, ako bolo popísané v kapitole s obmedzeniami robila leská vozovka (mokrú, ale klasifikovanú ako zasneženú). Priemerná doba výpočtu je 65 ms.

Tab. 6.2: Výsledky algoritmov analýzy vozovky na troch stanoviskách, hodnoty sú uvedené v %

Správna Určená	Suchá	Mokrú	Zasnežená
Suchá	100	0	0
Mokrú	0,68	92,57	6,75
Zasnežená	4,46	1,79	93,75

6.4.2 Testovanie prítomnosti hmly

Prítomnosť hmly bola otestovaná na 201 obrázkoch z rovnakých troch stanovísk. Na 116 obrázkoch je hmla a na 85 hmla nie je. Vzhľadom k tomu, že nie je dostatočne viditeľná obloha ani na jednom stanovisku, algoritmy som testoval ako binárnu klasifikáciu - je alebo nie je prítomná hmla. Vzhľadom k rôznorodosti stanovísk som zvolil jednotnú časť hornej časti obrázkov pre všetky stanoviská - horných 10 % riadkov obrázku. Priemerná doba výpočtu je 22 ms.

6.4.3 Testovanie detekcie zrážok

Prítomnosť sneženia bola z pohľadu analýzy pohybu otestovaná na 217 obrázkoch z rovnakých troch stanovísk. Zrážku sú prítomné na 167 obrázkoch, bez zrážok je

Tab. 6.3: Výsledky algoritmov detekcie hmly na troch stanoviskách, hodnoty sú uvedené v %

Správna		
Určená	Prítomná	Neprítomná
Prítomná	97,43	2,57
Neprítomná	0	100

50 obrázkov. Testovanie prítomnosti dažďa touto metódou som nepreviedol, lebo na obrázkoch nie sú dostatočné viditeľné padajúce kvapky. Priemerná doba výpočtu je 59 ms.

Tab. 6.4: Výsledky algoritmov detekcie zrážok na troch stanoviskách, hodnoty sú uvedené v %

Správna		
Určená	Prítomné	Neprítomné
Prítomné	92,26	7,74
Neprítomné	8,16	91,84

7 Použitie neurónových sietí

Táto časť práce je venovaná použitiu prístupu s obrovským rozvojom v úlohách počítačového videnia - neurónovej siete. Hlavným cieľom pridať neurónovú sieť do tejto práce bolo porovnanie klasických metód a metód s neurónovými sieťami.

7.1 Dataset a príprava dát

Pri práci s neurónovou sieťou je veľmi dôležitým krokom pripraviť dáta na samotné tréningovanie modelu a následnú validáciu a finálne testovanie natrénovaného modelu. Validáčnne dáta sú dáta, ktoré model nikdy nevidel a slúžia pre vyjadrenie úspešnosti modelu na "nových" dátach. Validáčnne dáta sú taktiež využívané v procese učenia. Celková úspešnosť modelu je testovaná na testovacích dátach. Zvolený pomer rozdelenia dát je 60:20:20, 60% dát datasetu je použitých na tréningovanie a zvyšných 20% na validáciu resp. 20% na finálne testovanie modelu. Obrázky boli najprv zmenšené na veľkosť 224x224 pixelov, následne náhodne premiešané, a až tak sa vytvoril tréningovací, validáčny a testovací dataset. Pre tento účel som vytvoril vlastný skript v jazyku Python.

Takto predpripravené dáta boli nahrané na môj Google Disk z toho dôvodu, že celý proces návrhu a tréningovania modelu prebiehal na platforme Google Colab. Google Colab ponúka užívateľom možnosť zadarmo využívať grafické karty Tesla K80 GPU po dobu 12 hodín. Po uplynutí tejto doby sú aj všetky nahraté dáta na Google Colab vymazané - z toho dôvodu som pracoval s Google Diskom, aby som nemusel dookola nahrávať značné množstvo obrazových dát. Výhoda použitia grafických kariet (GPU) oproti procesorom (CPU) je rádové skrátenie času potrebného na tréningovanie modelu.

Následne prebiehala augmentácia dát v tréningovej množine. Aj keď v prípade tejto práce je dostatočné množstvo dát (20 000), rozhodol som sa využiť možnosť augmentácie dát. Pod augmentáciou dát si možno predstaviť rôzne transformácie obrazu (zmenšenie, pootočenie, posunutie, osvetlenie), rozšíria dataset o relevantné dáta, čím dokážu zvýšiť celkovú robustnosť a invariantnosť modelu voči takýmto zmenám. Väčšina datasetu je prevzatá z odborných článkov, ktoré sa zaoberajú klasifikáciou počasia. Pre zvýšenie počtu obrázkov niektorých tried (hlavne hmla) som musel hľadať vhodné obrázky na internete a následne ich aj ručne anotovať. Dataset je tvorený kombináciou dopravnej scény a obecnej scény.

Tab. 7.1: Podoba použitého datasetu

Trieda	Trénovacie dáta	Validačné dáta	Testovacie dáta
Oblačno	6624	2207	2206
Hmla	2453	817	817
Dážď	2494	830	829
Sneženie	2248	749	748
Slnečno	5549	1850	1849

7.2 Výber architektúry

V tejto práci som vyskúšal natrénovať a porovnať dva rôzne typy neurónových sietí, a to konvolúčnu neurónovú sieť VGG16 a reziduálnu neurónovú sieť ResNet50.

Prenosové učenie (transfer learning)

V praxi je bežné, že sa neurónové siete netrénujú úplne od začiatku, ale sú použité už predtrénované modely na nejakom obecnom obrovskom datasete, ako napríklad ImageNet, ktorý obsahuje 1000 rôznych tried. Tento spôsob sa nazýva prenosové učenie (Transfer Learning). Existujú dva spôsoby, ako upraviť predtrénovaný model:

1. Feature Extraction využíva naučené reprezentácie na extrahovanie zmysluplných vlastností z nových vzoriek. Na predtrénovanom modeli zmeníme poslednú vrstvu (klasifikátor, posledná, plne-prepojená vrstva) a iba on bude trénovaný od začiatku. Váhy v predchádzajúcich vrstvách ostanú zafixované resp. zamrazené.
2. Fine-tuning namiesto tréningu iba nového klasifikátora, sú rozmrazené aj niektoré vrstvy nad samotným klasifikátorom. To znamená, že počas tréningu sa menia aj váhy rozmrazených vrstiev počas spätného šírenia.

V tejto práci som vyskúšal obe varianty. Všetky návrhy modelov, tréningu modelov aj vykreslenia výsledkov boli prevedené v Pytorch. Jednotlivé použité skripty vychádzajú z [53].

7.3 Výber architektúry

Ako prvú architektúru som vyskúšal ResNet50, ktorej som zmrazil všetky vrstvy okrem mnou pridaného klasifikátora. Môj klasifikátor obsahuje ReLU aktivačnú funkciu, Dropout vrstvu, lineárnu funkciu a na záver Softmax aktivačnú funkciu. Všetky

vyskúšané modely majú rovnaký klasifikátor, rovnaký krok učenia (learning rate = 0,01), veľkosť dávky (batch-size = 32), optimalizátor (SGD) a stratovú funkciu (CrossEntropy). Parameter learning rate vyjadruje, aký veľký krok sa má spraviť počas každého cyklu tréningovania, aby sa čo najviac minimalizovala hodnota stratovej funkcie. Optimalizátor je pomocný algoritmus, ktorý úpravou váh vypomáha znížiť hodnoty stratovej funkcie. Jednotlivé tréningovania netrvali rovnaké počty epoch (cyklov), lebo po dosiahnutí zadaného počtu epoch, kedy nenastalo žiadne zlepšenie úspešnosti, je tréningový proces ukončený.

Hlavným cieľom Dropout vrstvy je zníženie šance, že sa model pretrénuje a funguje tak, že počas tréningovania sa určité percento náhodne vybraných neurónov deaktivuje - nastaví na nulu. Lineárna funkcia má za úlohu nastaviť požadovaný počet výstupných tried, ktoré má model klasifikovať. Predtrénované modely majú väčšinou nastavených 1000 výstupných tried ako defaultnú hodnotu klasifikácie. Softmax som použil preto, lebo detekcia počasia je úloha, ktorá ma na výstupe viaceré triedy, v tejto práci uvažujem 5 tried - Oblačno, Hmlisto, Dážď, Sneženie a Slniečno. Softmax umožní, aby každá trieda mala na výstupe priradenú odhadovanú pravdepodobnosť, že daný obrázok do nej patrí. Súčet pravdepodobností musí byť rovný 1.

Prvý pokus testovania, ako som spomenul, bol zameraný na využitie predtrénovaného modelu ResNet50 so zamrazením všetkých vrstiev okrem môjho klasifikátora.

Druhý pokus testovania bol zameraný opäť na využitie predtrénovaného modelu ResNet50, ale tentokrát som okrem môjho klasifikátora trénoval aj 3 vrstvy nad ním.

Tretí a posledný s architektúrou ResNet bol zameraný na tréningovanie komplet celého modelu ResNet50 od začiatku - nie je zamrazená žiadna vrstva.

Ako posledný pokus použitia neurónovej siete v tejto práci som vyskúšal natréňovať architektúru CNN, konkrétne VGG16.

Všetky pokusy boli tréňované na menšom datasete, ktorý obsahuje 1000 obrázkov v každej triede, a až "najlepší" z nich bol natréňovaný na kompletnom datasete.

7.4 Dosiahnuté výsledky

Uvedená tabuľka 7.2 zobrazuje výsledné úspešnosti modelov, ktoré som natréňoval. Prvé štyri riadky sú modely natréňované na zmenšenom datasete a posledný riadok je finálny model, natréňovaný na kompletnom datasete. V tejto kapitole je zobrazený graficky iba výsledok finálneho modelu, zvyšné grafy sú uvedené v prílohe. Z výsledkov je jasné, že najhoršie dopadol model, ktorý nemal zamrazenú ani jednu vrstvu. Vykazuje najmenšiu percentuálnu úspešnosť a zároveň výrazné pretrénovanie (over-fitting). Pretrénovanie modelu sa pozná tak, že validačná strata je výrazne

Tab. 7.2: Dosiahnuté výsledky použitím neurónových sietí

Typ siete	Trénovacia presnosť	Trénovacia strata	Validačná presnosť	Validačná strata
Klasifikátor ResNet50	~78 %	~0,60	~80 %	~0,55
Klasifikátor + 3 vrstvy ResNet50	~80 %	~0,55	~83 %	~0,55
Všetky vrstvy ResNet50	~74 %	~0,84	~68 %	~0,74
Klasifikátor VGG16	~79 %	~0,41	~84 %	~0,59
Klasifikátor + 3 vrstvy ResNet50	~83 %	~0,52	~81 %	~0,54

vyššia ako trénovacia chyba. To znamená, že model nesprávne zovšeobecňuje neznáme dáta. Pretrénovanie sa nepodarilo odstrániť ani dropout vrstvou.

Problémom pretrénovania trpí rovnako aj model VGG16, kde taktiež nepomohlo zaradenie dropout vrstvy. Ďalšou nevýhodou je, že trénovanie trvalo tomuto modelu najdlhšie, cca. 2 hodiny. Na druhej strane, VGG16 dosiahla najvyššiu validačnú presnosť zo všetkých použitých modelov.

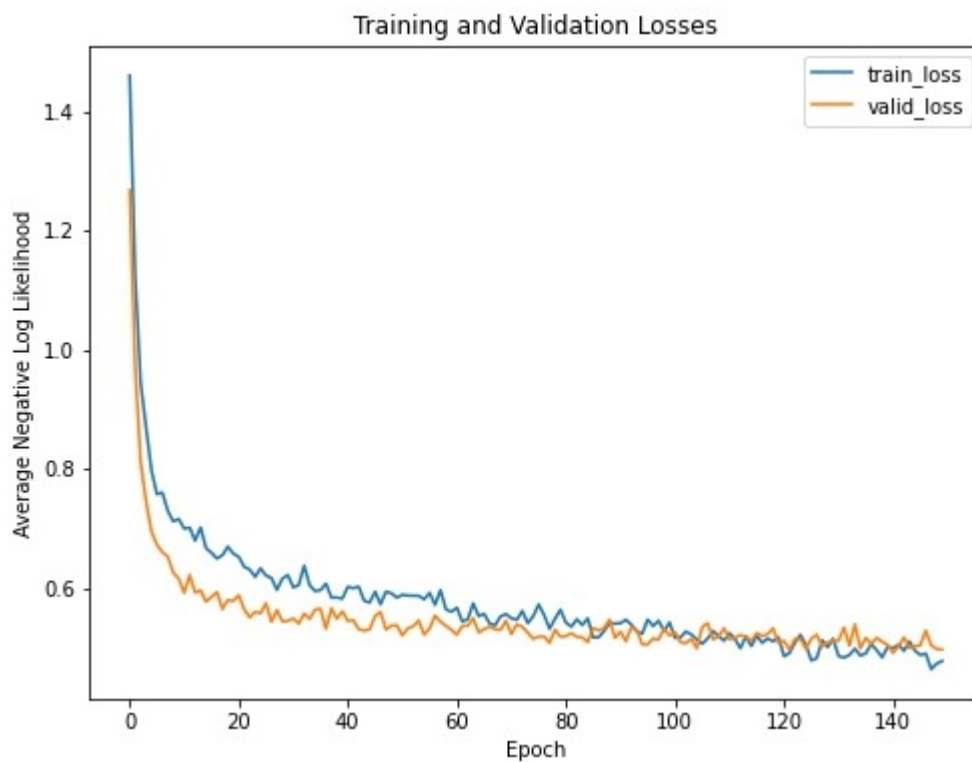
Najlepšou voľbou z testovaných modelov sa javí model, ktorý má odmrazených posledných pár vrstiev, a tým môže byť dosiahnuté určité zlepšenie úspešnosti, aj keď v tomto prípade je zlepšenie malé. Všetky ResNet modely sa trénovali cca. 1,5 hodiny.

Pre finálne trénovanie modelu na celom datasete som na základe dosiahnutých úspešností na čiastočnom datasete zvolil model ResNet50, ktorý má odmrazené posledné 3 vrstvy. Celkové dosiahnuté výsledky tohto modelu sú znázornené v obrázku s confusion maticou a tabuľke celkovej štatistiky modelu. Celková presnosť tohto modelu je približne 80 %. Doba trénovania trvala približne 5,5 hodín s využitím grafického akcelarátoru.

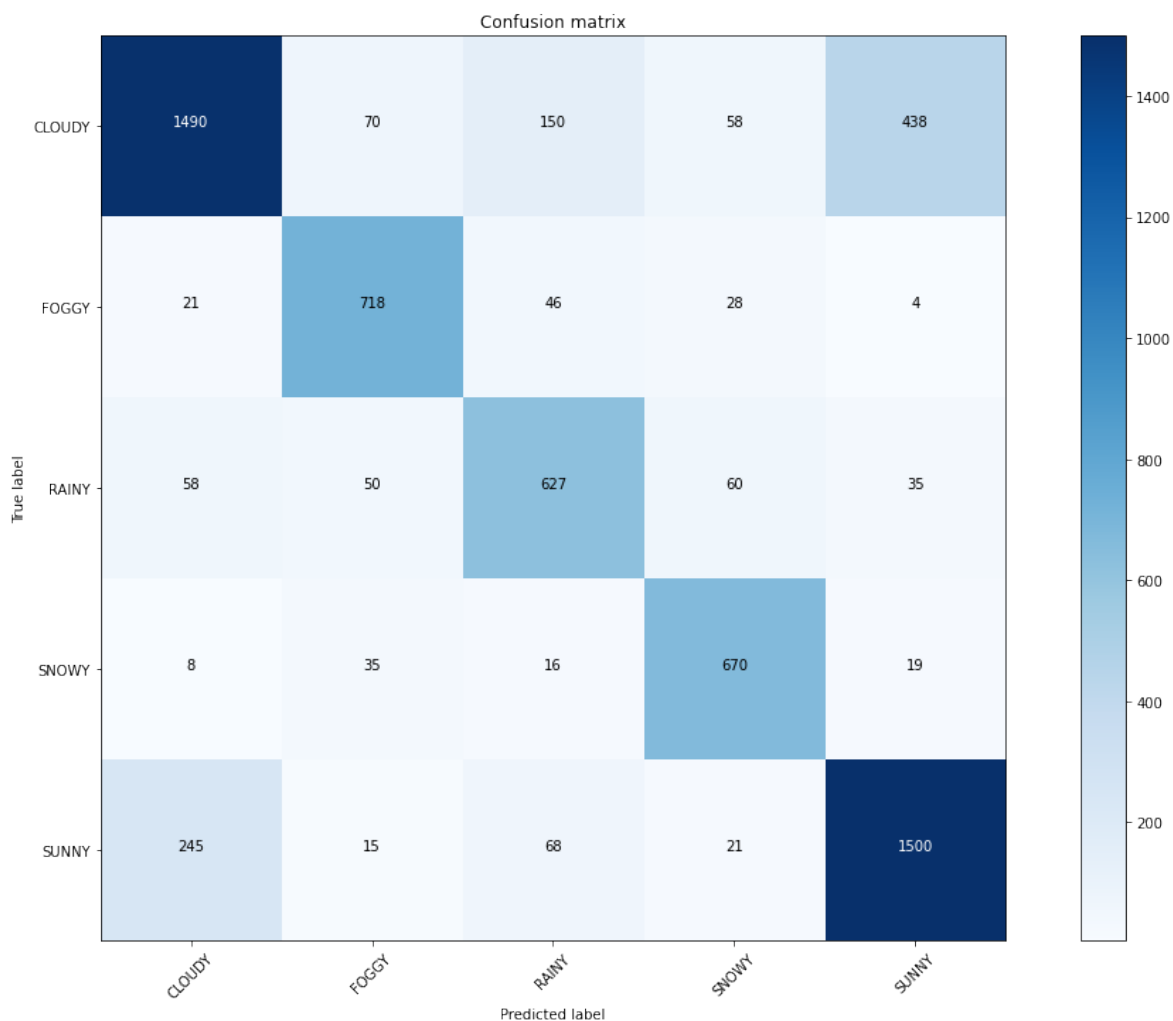
Tabuľka 7.3 znázorňuje bližšiu štatistiku úspešnosti finálneho modelu. V úlohe klasifikácie počasia si treba uvedomiť, že môžu nastať aj viaceré stavy naraz (hmla a sneženie, hmla a dážď). Celková úspešnosť modelu vyšla na 80,3 %, ale ak zoberieme do úvahy fakt, že môžu nastať aj dva stavy počasia súčasne, je vidno rapídny nárast úspešnosti detekcie, vo všetkých triedach presiahla hranicu 90 %. Stĺpec Top 2 berie



Obr. 7.1: Grafický výstup natrénovaného modelu na celom datasete



Obr. 7.2: Grafický výstup presností natrénovaného modelu na celom datasete



Obr. 7.3: Grafický výstup strát natrénovaného modelu na celom datasete pomocou confusion matice

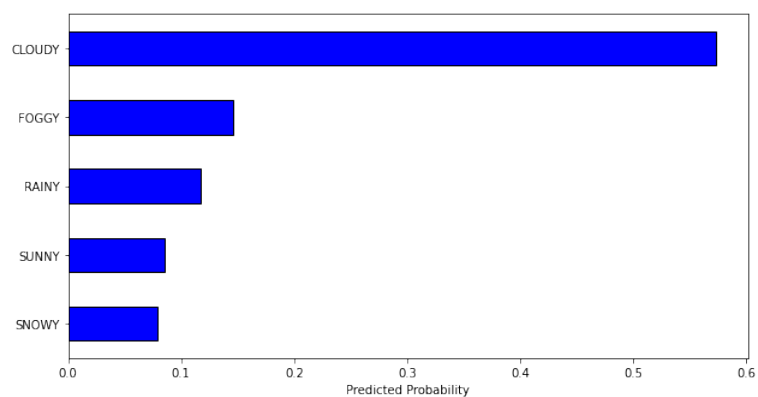
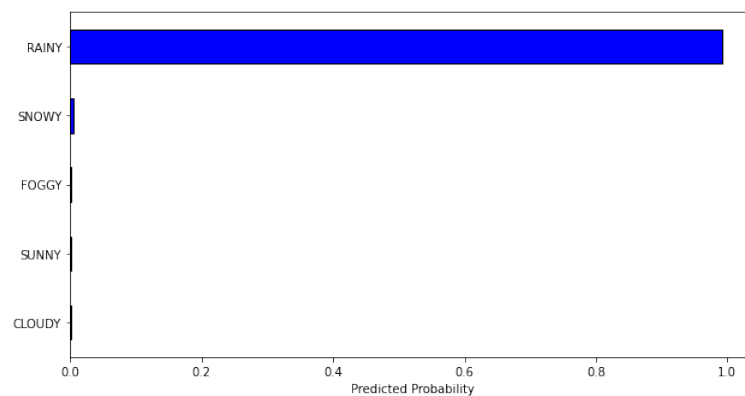
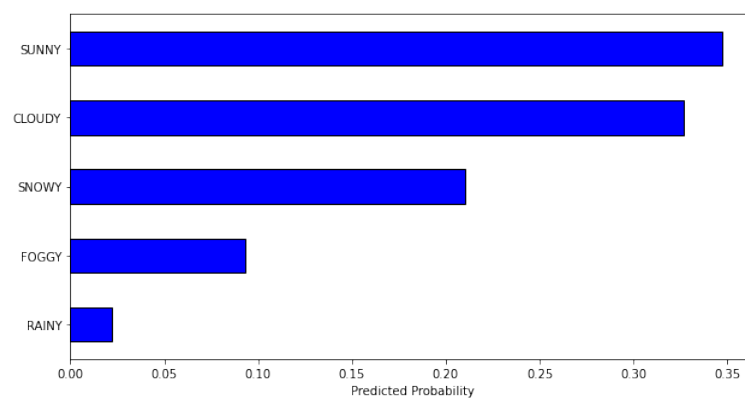
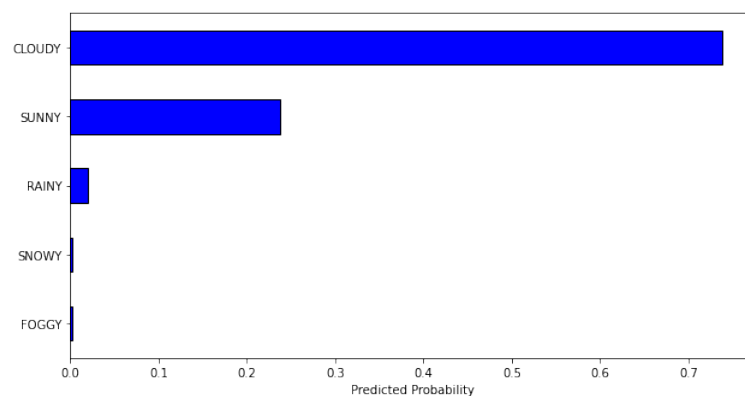
Tab. 7.3: Štatistika finálneho modelu

Trieda	Top 1	Top 2	Top 3	Strata
Oblačno	67,5 %	93,8%	97,8 %	0,79
Hmla	87,9 %	96,1 %	98,4 %	0,37
Dážď	75,5 %	90,4 %	96,9 %	0,68
Sneženie	89,6 %	97,6%	98,8 %	0,3
Slnečno	81,1 %	94,4 %	98,4 %	0,49

ako správnu klasifikáciu aj tú možnosť, keď model síce najvyššiu pravdepodobnosť určí nepresne, ale druhá najvyššia pravdepodobnosť je už správna.

Bližšiu štatistiku ohľadne výstupných tried zobrazuje obrázok 7.3. Medzi najčastejšie zámeny došlo pri stave oblačno (cloudy) a slnečno (sunny), čo je síce chyba, ale v prípade tejto práce nie je zásadná, v tomto prípade sú zaujímavé hlavne extrémny počasia, ktoré majú vplyv na dopravu. Rovnako ako pri klasických metódach, je najhoršia úspešnosť detekcie dažďa. Problematika detekcie dažďa na obrázkoch je popísaná v predchádzajúcej časti práce.

Výstup odhadu pravdepodobností modelom možno vidieť na obrázku 7.4. Prvý a posledný obrázok zobrazujú, ako nastali súčasne viaceré stavy počasia naraz (slnečno a oblačno resp. oblačno, hmla a dážď). V oboch prípadoch sú prítomným stavom priradené najvyššie pravdepodobnosti. Vyhodnotenie pravdepodobnosti jedného obrázku (224 x 224) trvá menej ako jednu sekundu.



Obr. 7.4: Grafický výstup odhadu pravděpodobnosti modelom

8 Porovnanie klasických metód a neurónových sietí

Vybrať definitívne lepšiu metódu je veľmi náročná úloha. Každá z nich má výhody aj nevýhody. Najdôležitejším faktorom výberu je zváženie, o akú scénu sa jedná, aké je presné zadanie úlohy, koľko výstupov musím uvažovať, koľko dát mám dostupných a podobne, a na základe toho vybrať metódu použitia. Čím sa javí úloha komplexnejšie, tým je vhodnejšie použitie neurónovej siete.

Výhodou klasických metód je, že môžu byť veľmi jednoducho prispôsobené rôznym potrebám úlohy, pre ktoré sú určené. Na druhej strane, čím je scéna rozamnitejšia a celkovo zložitejšia, tým sa vhodnosť klasických metód rapídne znižuje. Ďalšou výhodou je minimálna výpočetná a pamäťová náročnosť v porovnaní s neurónovými sieťami. Ako najväčšia nevýhoda klasických metód voči neurónovým sieťam v tejto práci je nutnosť referenčného obrázka, podľa ktorého je prevedená klasifikácia (je potrebná sekvencia obrázkov).

Neurónové siete majú podstatne širší rozsah použitia, ale prispôsobenie na mieru konkrétnym a špecifickým úlohám je náročnejšie ako pri klasických metódach. Neurónové siete sú ako black-box, veľmi ťažko interpretovateľné, prečo sa niektoré oblasti v obraze aktivovali, a iné nie. Neurónové siete vďaka rýchlemu vývoju a pokroku, stále s lepšími výsledkami v čoraz náročnejších úlohách dávajú jasne najavo, že ich horná hranica možností je ďaleko vyššia ako hranica klasických metód. Na druhú stranu, stále existujú problémy, na ktoré sa javí nasadenie neurónovej siete ako overkill. Veľkou výhodou neurónových sietí je ich presnosť, s ktorou dokážu vyriešiť problémy počítačového videnia. Neurónová sieť nepotrebuje žiadnu referenciu pre vyhodnotenie výstupu.

Ako najlepšia možnosť sa javí kombinácia oboch prístupov, každá metóda má svoje výhody. Preto je najlepšie využiť viacero metód ako kombináciu výstupov, ktoré sa navzájom dopĺňajú, potvrdzujú alebo vylučujú.

8.1 Logika vyhodnocovania

Táto práca je zameraná skôr na vytvorenie prototypov jednotlivých metód. V prípade nasadenia do reálneho provozu je nutné vyriešiť logiku vyhodnocovania. Stav počasia sa vo väčšine prípadov mení postupne, rádovo v minútach. Aj keď sú implementované metódy výpočetne pomerne rýchle, určite nemá význam vyhodnocovať stav každú sekundu. Za vhodný interval vyhodnotenia považujem interval medzi 3 až 5 minútami. Počas tohto intervalu môžu byť vypočítané dočasné výsledky, napr. každých 30 sekúnd a finálny výsledok bude kombináciou (napr. priemernou

hodnotou) všetkých pomocných výsledkov za stanovený interval. Určenie stavu z viacerých menších časových okamžikov môže zvýšiť presnosť klasifikácie skutočného stavu počasia.

Záver

Cielom tejto diplomovej práce bolo zoznámenie sa s úlohou rozpoznávanie počasia z obrázkov, naštudovať existujúce riešenia, navrhnúť možné prístupy a otestovať riešenie.

Prvá kapitola je venovaná popisu počasia - základné elementy, spôsob a nástroje pre ich meranie bez použitia kamerových dát. Hlavnou motiváciou analýzy počasia z kamerových obrázkov je, čo do najväčšej možnej miery nahradiť drahé meteorologické stanice kamerami. V dnešnej dobe sú kamery bežnou súčasťou nielen rôznych strategických dopravných uzlov ale aj bežných ciest. Koniec kapitoly je venovaný vážnosti vplyvu nepriaznivého počasia na dopravu z ekonomického hľadiska, ale aj z hľadiska bezpečnosti ľudí.

Druhá kapitola je venovaná teoretickému popisu základných častí klasických metód počítačového videnia, ktoré som sám využil v praktickej časti. V tretej kapitole je stručne priblížená problematika použitia neurónových sietí, z ktorej opäť čerpám v praktickej časti.

Štvrtá kapitola je venovaná získaniu čo najviac relevantných informácií o problematike detekcie počasia z obrazových dát. Sú popísané viaceré možné prístupy a taktiež aj bližšie popísané niektoré konkrétne návrhy. Na základe tejto kapitoly som vytvoril návrh algoritmov, ktoré som vyskúšal implementovať a otestovať.

Piata kapitola popisuje vstupné dáta, ktoré som vyzbieral a zostavil datasety, s ktorými som následne pracoval. Datasety možno rozdeliť do troch skupín - dataset so stavom vozovky, dataset počasia s obecnou scénou a dataset počasia s dopravnou scénou.

Úvod šiestej kapitoly rozoberá možné prístupy, ako vysegmentovať vozovku, ako následne určiť jej stav, ako detekovať prítomnosť hmly, poprípade ju rozlíšiť od iných stavov počasia a nakoniec, ako detekovať pohyb vplyvom zrážok. Keďže počasie nie je konkrétny objekt, ale skôr abstraktná vec, pokúsil som sa založiť algoritmy na základoch, ako človek rozlišuje alebo vníma zmenu počasia. Uvažoval som hlavne zmenu farebného rozloženia, zmenu kontrastu, zmenu počtu hrán, zmenu vo frekvenčnom obsahu obrázku a podobne. Po návrhu nasledovala fáza testovania, všetky navrhnuté klasické metódy som otestoval na troch reálnych dopravných stanoviskách. Podarilo sa mi analyzovať tri stavy vozovky - suchá, mokrá, zasnežená vrátane parametrizácie, koľko vozovky je pokrytej snehom/vodou. Taktiež sa mi podarilo úspešne detekovať prítomnosť hmly a odhad viditeľnosti. Na záver sa podarila aj detekcia sneženia a parametrizácia jeho intenzity. Jediná vec, ktorá sa nepodarila priamo detekovať klasickými metódami, je detekcia dažďa. Dážď bola najťažšia časť detekcie hlavne aj preto, lebo na väčšine testovacích fotiek neboli viditeľné dažďové kvapky. Nakoniec som prítomnosť dažďa určil nepriamo z výsledku analýzy stavu vozovky.

Všetky popísané klasické metódy, ktoré malo zmysel k charakteru testovacieho datasetu použiť, dosiahli úspešnosť na testovacom datasete nad 90 %.

Ako druhý, úplne odlišný spôsob detekcie zmien počasia, som použil neurónovú sieť, ktorej je venovaná siedma kapitola. Uvažoval som päť rôznych stavov počasia, ktoré môžu nastať - oblačno, hmlisto, dážď, sneženie a slnečno. Najprv bolo nutné pripraviť dáta na fázu trénovania modelu. Použitý dataset má vyše 20000 fotiek zachytávajúcich počasie s rôznymi scénami. Finálny model má úspešnosť klasifikácie približne 80 %. Úspešnosť klasifikácie výrazne znížil fakt, že na jednom obrázku môžu byť zobrazené aj viaceré triedy súčasne. V prípade, že je klasifikácia akceptovaná za správnu aj v prípade, že sa správna trieda nachádza medzi dvomi výstupnými triedami s najvyššou pravdepodobnosťou, celková úspešnosť vzrástla na hodnotu 94,5 % s tým, že všetky triedy sú nad hranicou 90 %. V závere práce je popísané porovnanie použitia neurónovej siete a použitia klasických metód a možný návrh logiky vyhodnotenia v reálnom dopravnom provoze.

Literatúra

- [1] ČSN ISO 690 (01 0197) *Guide to Instruments and Methods of Observation* Geneva: World Meteorological Organization (WMO) , 2018 edition. [online] [cit. 3. 1. 2021]. <https://library.wmo.int/index.php?id=12407&lvl=notice_display#.X_Jq3thKiUk/>.
- [2] National Geographic Encyclopedia [online] [cit. 3. 1. 2021]. <<https://www.nationalgeographic.org/encyclopedia/weather/>>.
- [3] Britannica Students [online] [cit. 3. 1. 2021]. <<https://kids.britannica.com/students/article/weather/277681>>.
- [4] The Elements Of Weather And Climate And The Instruments Measuring Them [online] [cit. 3. 1. 2021]. <<https://ownyourweather.com/elements-of-the-weather-and-climate/>>.
- [5] What Is Wind, How Are Winds Formed, And What Are The Different Types Of Winds? [online] [cit. 3. 1. 2021]. <<https://ownyourweather.com/what-is-wind-and-how-are-winds-formed/>>.
- [6] What Is The Coriolis Effect And How Does It Affect Our Weather? [online] [cit. 3. 1. 2021]. <<https://www.lufft.com/products/road-runway-sensors-292/non-invasive-road-sensor-nirs31-umb-2307/>>. <<https://ownyourweather.com/what-is-the-coriolis-effect/>>.
- [7] Pyrometer [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.lufft.com/products/road-runway-sensors-292/non-invasive-road-sensor-nirs31-umb-2307/>>.
- [8] NTC termistor [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.lufft.com/products/road-runway-sensors-292/intelligent-passive-road-sensor-irs31pro-umb-2306/>>.
- [9] Anemometer [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.environmental-expert.com/products/ultrasonic-anemometer-2d-116845>>.
- [10] Snímač vlhkosti [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.lufft.com/products/compact-weather-sensors-293/ws300-umb-smart-weather-sensor-1850/>>.

- [11] Zrážkomer [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.netatmo.com/en-us/weather/weatherstation/raingauge>>.
- [12] Doppler [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.lufft.com/products/compact-weather-sensors-293/ws600-umb-smart-weather-sensor-1832/>>.
- [13] Transmisometer [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.vaisala.com/en/products/instruments-sensors-and-other-measurement-devices/weather-stations-and-sensors/lt31>>.
- [14] Dopravná meteostanica [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://www.microstep-mis.com/drupal/web/sites/default/files/2018-10/road_systems_web.pdf>.
- [15] Statistika nehodovosti [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.policie.cz/clanek/clanek/statistika-nehodovosti-872434.aspx>>.
- [16] Zheng Zhang, Huadong Ma. *Multi-class weather classification on single images* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://mwidataset.weebly.com/>>.
- [17] Zheng Zhang, Huadong Ma., et al. *Scene-free multi-class weather classification on single images. Neurocomputing, 2016, 207: 365-373.* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://mwidataset.weebly.com/>>.
- [18] *AAU RainSnow Traffic Surveillance Dataset* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.kaggle.com/aalborguniversity/aau-rainsnow>>.
- [19] Milan Sonka, Václav Hlavác, Roger Boyle *Image processing, Analysis, and Machine Vision* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.amazon.com/Image-Processing-Analysis-Machine-Vision/dp/049508252X>>.
- [20] Ing. Karel Horák, Ph.D. *Detekce hran a rohů* [online] [cit. 3. 1. 2021]. Dostupné z URL: <http://vision.uamt.feec.vutbr.cz/ZVS/lectures/08_Detekce_hran_a_rohu.pdf>.
- [21] Rafael C. Gonzalez, Richard E. Woods *Digital Image Processing, FOURTH EDITION.* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.amazon.com/Digital-Image-Processing-Rafael-Gonzalez/dp/9353062985>>.
- [22] Dilpreet Kaur et al *Various Image Segmentation Techniques, International Journal of Computer Science and Mobile Computing, Vol.3 Issue.5, May- 2014,*

- pg. 809-814 [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ijcsmc.com/docs/papers/May2014/V3I5201499a84.pdf>>.
- [23] Sumeet Bansal *What is Image Segmentation?* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.analytixlabs.co.in/blog/what-is-image-segmentation/>>.
- [24] OpenCV docs *Image Segmentation with Watershed Algorithm* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://docs.opencv.org/master/d3/db4/tutorial_py_watershed.html>.
- [25] Road Segmentation by Watershed Algorithms *S.BEUCHER,M.BILODEAU,X.YU. Centre de Morphologie Mathématique, ENSMPFontainebleau, France* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://www.researchgate.net/publication/230837974_Road_segmentation_by_watershed_algorithms>.
- [26] Soget Lee *Lines Detection with Hough Transform* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://towardsdatascience.com/lines-detection-with-hough-transform-84020b3b1549>>.
- [27] Nushaine Ferdinand *A Deep Dive into Lane Detection with Hough Transform* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://towardsdatascience.com/a-deep-dive-into-lane-detection-with-hough-transform-8f90fdd1322f>>.
- [28] Ing. Karel Horák, Ph.D. *Matematická morfologie*. [online] [cit. 3. 1. 2021]. Dostupné z URL: <http://vision.uamt.feec.vutbr.cz/ZVS/lectures/11_Matematicka_morfologie.pdf>.
- [29] OpenCV docs *Morphological Transformations* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://docs.opencv.org/master/d9/d61/tutorial_py_morphological_ops.html>.
- [30] *Introduction to Fourier Transforms for Image Processing* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.cs.unm.edu/~brayer/vision/fourier.html>>.
- [31] *A Gentle Introduction To Neural Networks Series — Part 1* [online] [cit. 22. 4. 2021]. Dostupné z URL: <<https://towardsdatascience.com/a-gentle-introduction-to-neural-networks-series-part-1-2b90b87795bc>>.

- [32] *Drawing neural network with tikz* [online] [cit. 22. 4. 2021]. Dostupné z URL: <<https://tex.stackexchange.com/questions/153957/drawing-neural-network-with-tikz>>.
- [33] *Activation Functions in Neural Networks* *Activation Functions in Neural Networks* [online] [cit. 22. 4. 2021]. Dostupné z URL: <<https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>>.
- [34] *Understanding Neural Networks. From neuron to RNN, CNN, and Deep Learning* [online] [cit. 22. 4. 2021]. Dostupné z URL: <https://medium.com/analytics-vidhya/understanding-neural-networks-from-neuron-to-rnn-cnn-and-deep-learning-cd88e90e0a90>
- [35] *CNN vs. RNN vs. ANN – Analyzing 3 Types of Neural Networks in Deep Learning* [online] [cit. 22. 4. 2021]. Dostupné z URL: <https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/>
- [36] *Difference between AlexNet, VGGNet, ResNet, and Inception* [online] [cit. 22. 4. 2021]. Dostupné z URL: <<https://towardsdatascience.com/the-w3h-of-alexnet-vggnet-resnet-and-inception-7baaaecccc96/>>.
- [37] *An Overview of ResNet and its Variants* [online] [cit. 22. 4. 2021]. Dostupné z URL: <<https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035/>>.
- [38] Jérémie Bossu, Nicolas Hautiere, Jean-Philippe Tarel *Rain or Snow Detection in Image Sequences Through Use of a Histogram of Orientation of Streaks* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://www.researchgate.net/publication/220660048_Rain_or_Snow_Detection_in_Image_Sequences_Through_Use_of_a_Histogram_of_Orientation_of_Streaks>.
- [39] Dong Huiying, Zhao Xuejing *Detection and Removal of Rain and Snow from Videos Based on Frame Difference Method* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/7162843>>.
- [40] S. Bronte, L. M. Bergasa, P. F. Alcantarilla *Fog detection system based on computer vision techniques* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/5309842>>.
- [41] Mohamed R Ibrahim, James Haworth, T. Cheng *WeatherNet: Recognising Weather and Visual Conditions from Street-Level Images*

- Using Deep Residual Learning* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://www.researchgate.net/publication/337664561_WeatherNet_Recognising_Weather_and_Visual_Conditions_from_Street-Level_Images_Using_Deep_Residual_Learning>.
- [42] Andrea Lagorio, Enrico Grosso, Massimo Tistarelli *Automatic Detection of Adverse Weather Conditions in Traffic Scenes* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/4730423>>.
- [43] Mihai Negru, Sergiu Nedevschi *Image based fog detection and visibility estimation for driving assistance systems* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/6646102?section=abstract>>.
- [44] Mario Pavlic, Heidrun Belzner, Gerhard Rigoll, Slobodan Ilic *Image based fog detection in vehicles* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://www.researchgate.net/publication/261155082_Image_based_fog_detection_in_vehicles>.
- [45] Aurélien Cord, Didier Aubert *Towards rain detection through use of in-vehicle multipurpose cameras* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/5940484>>.
- [46] Martin Roser, Frank Moosmann *Classification of weather situations on single color images* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://ieeexplore.ieee.org/document/4621205>>.
- [47] Zheng Zhang, Huadong Ma, Huiyuan Fu, Cheng Zhang *Scene-free multi-class weather classification on single images* [online] [cit. 3. 1. 2021]. Dostupné z URL: <https://web.cse.ohio-state.edu/~zhang.7804/Cheng_NC2016.pdf>.
- [48] Bin Zhaoa, Xuelong Lib, Xiaoqiang Lub, Zhigang Wanga *A CNN-RNN Architecture for Multi-Label Weather Recognition* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://arxiv.org/pdf/1904.10709.pdf>>.
- [49] Sen-ching S. Cheung, Chandrika Kamath *Robust techniques for background subtraction in urban traffic video* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5308/0000/Robust-techniques-for-background-subtraction-in-urban-traffic-video/10.1117/12.526886.short?SS0=1>>.
- [50] Sen-ching S. Cheung, Chandrika Kamath *Robust techniques for background subtraction in urban traffic video* [online] [cit. 3. 1. 2021]. Dostupné z URL: <<https://www.spiedigitallibrary.org/conference-proceedings-of-spie/5308/0000/Robust-techniques-for-background-subtraction-in-urban-traffic-video/10.1117/12.526886.short?SS0=1>>.

spiedigitallibrary.org/conference-proceedings-of-spie/5308/0000/
Robust-techniques-for-background-subtraction-in-urban-traffic-video/
10.1117/12.526886.short?SS0=1>.

[51] *Přehled webových kamer ČHMÚ* [online] [cit. 3. 1. 2021]. Dostupné z URL:
<<https://www.chmi.cz/files/portal/docs/meteo/kam/>>.

[52] *Wikipédie, stránka byla naposledy editována 18. 2. 2021* [online] [cit. 22. 4. 2021].
Dostupné z URL: <<https://www.chmi.cz/files/portal/docs/meteo/kam/>>.

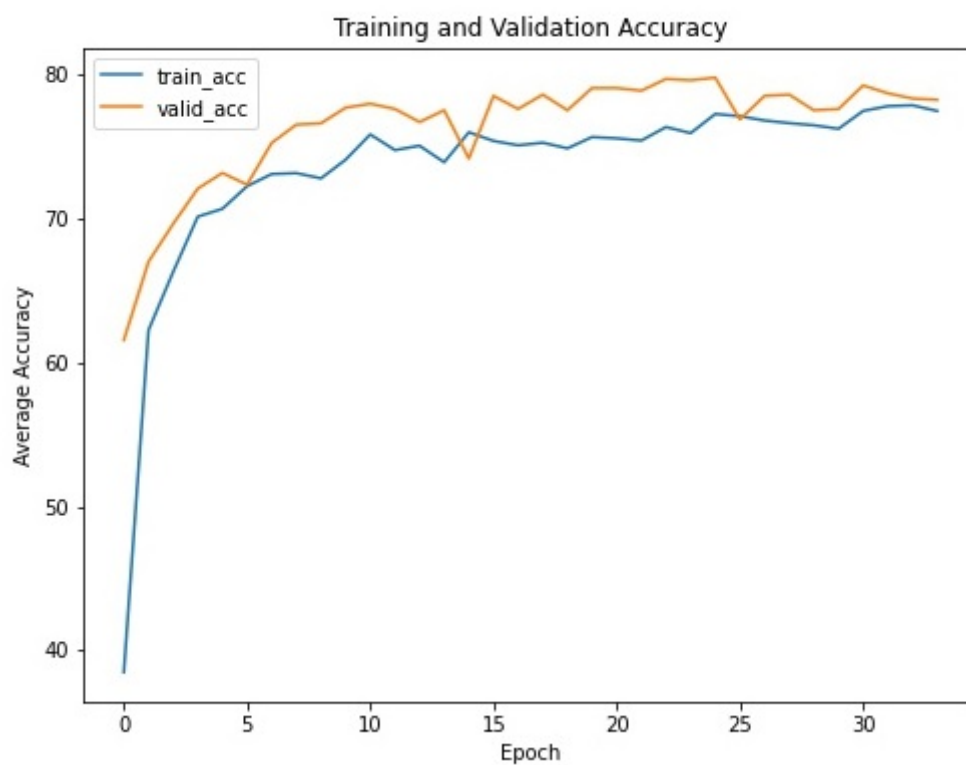
[53] *End to End Pipeline for setting up Multiclass Image Classification for Data Scientists* [online] [cit. 22. 4. 2021].
Dostupné z URL: <<https://towardsdatascience.com/end-to-end-pipeline-for-setting-up-multiclass-image-classification-for-data->

Zoznam príloh

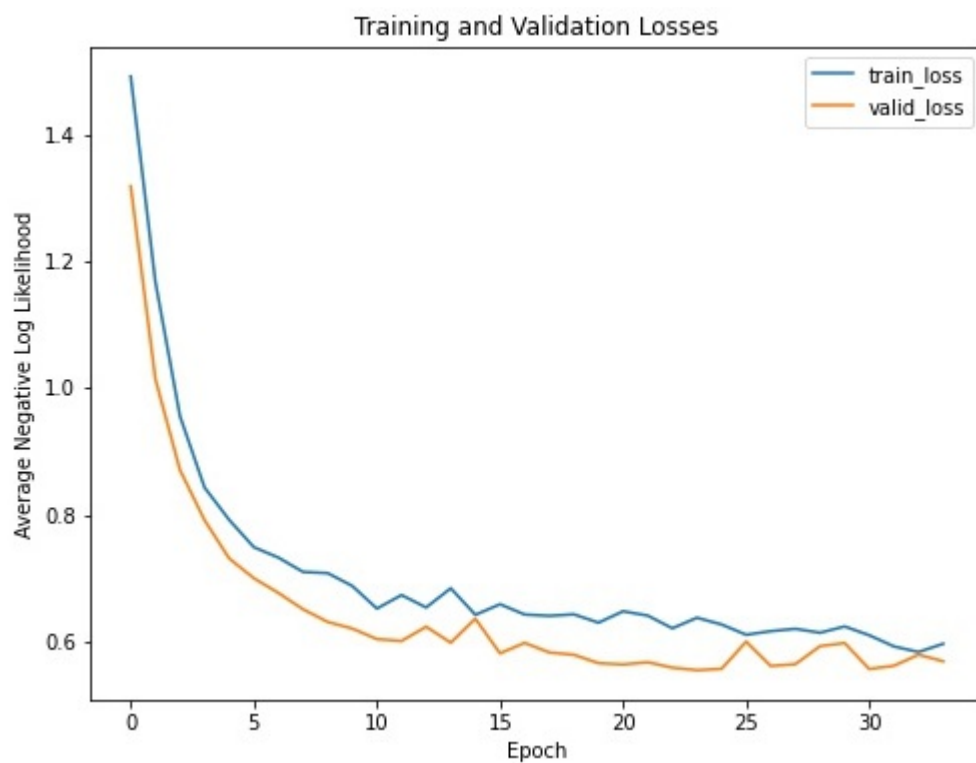
A Výsledné grafy použitých modelov

109

A Výsledné grafy použitých modelov



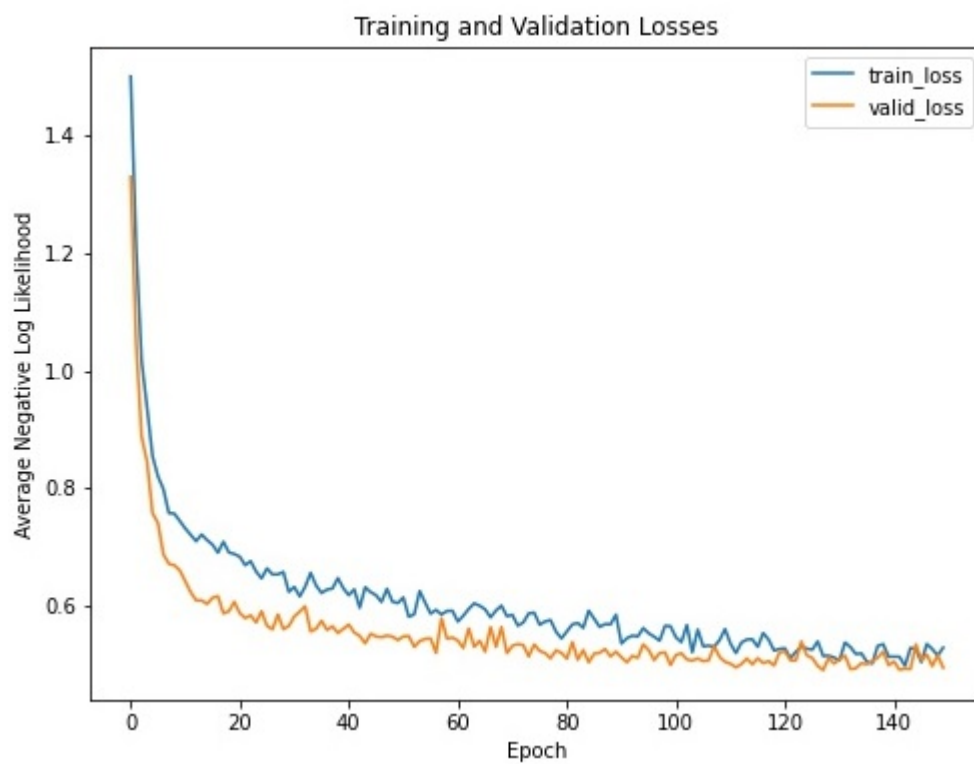
Obr. A.1: Klasifikátor ResNet50



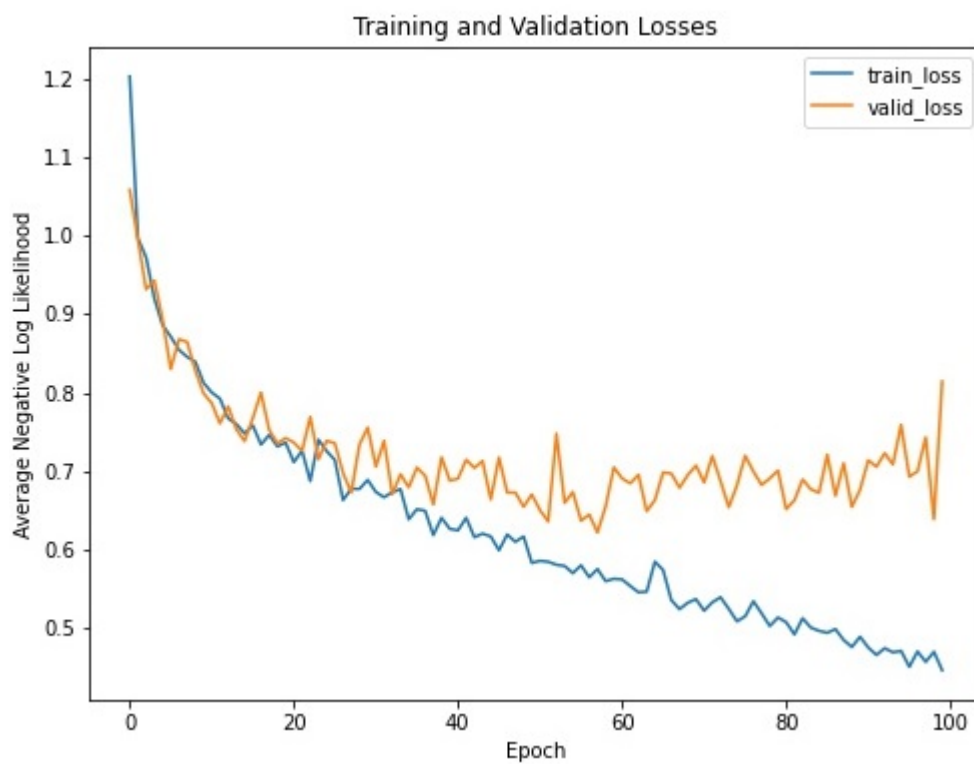
Obr. A.2: Klasifikátor ResNet50



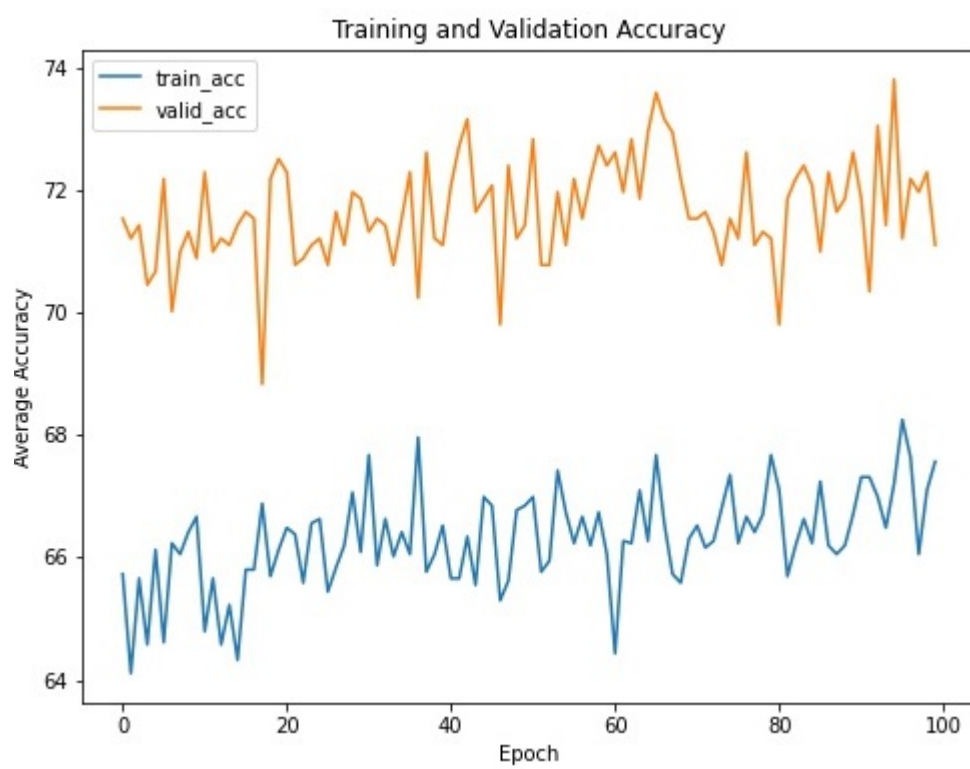
Obr. A.3: Klasifikátor + 3 vrstvy ResNet50



Obr. A.4: Klasifikátor + 3 vrstvy ResNet50



Obr. A.5: Klasifikátor VGG16



Obr. A.6: Klasifikátor VGG16